

Thesis submitted for the degree of

Doctor of Philosophy in Computer Architecture

by

Joan CAPDEVILA PUJOL

# Exploring the Topical Structure of Short Text through Probability Models: from Tasks to Fundamentals

supervised by

Jordi TORRES VIÑALS

Professor, Polytechnic University of Catalonia & Barcelona Supercomputing Center

Jesús CERQUIDES BUENO

Tenured Researcher, Artificial Intelligence Research Center

June 25, 2019



UNIVERSITAT POLITÈCNICA  
DE CATALUNYA  
BARCELONATECH



**Barcelona  
Supercomputing  
Center**  
*Centro Nacional de Supercomputación*



*Ancora Imparo*



# Acknowledgements

I can barely start reading any dissertation without first looking at the acknowledgements section to know about the journey and people who made it possible. My Ph.D. was no different and it would not have been possible without the help and support of many. Apart from those mentioned below, I want to recall anyone who has awakened my scientific interest in the past years, instilled persistence and purpose throughout the journey and remembered me to always enjoy the ride. We certainly did it and I have no words to describe how grateful I feel that our paths have crossed at this stage of my life.

I would like to express my deepest gratitude to my advisors and mentors Dr. Jesús Cerquides and Prof. Jordi Torres for giving me the life-changing opportunity of pursuing a Ph.D. under their guidance and the support throughout the whole process. Among many other things, I am extremely grateful to Prof. Torres for his encouragement and trust in my application to “La Caixa” fellowship, his constant help to identify my research interests and his flexibility to integrate them into his research group. I cannot thank Dr. Cerquides enough for accepting me as his student, teaching me the ins and outs of research and putting me in touch with his network. I often remember Jesús’ words on my first day with him: “Joan, don’t worry, we’ll have fun”. We certainly did it, but we have also worked hard and learnt a lot. I feel very proud of my mentors for what and how they have done it.

I cannot find the words to thank Dr. François Petitjean, Prof. Wray Buntine and Prof. Geoff Webb for inviting me to the Machine Learning group at Monash University as a visiting researcher and their inspiring mentorship. Not only did they helped me to develop further the theoretical foundations of my thesis, but they encouraged me to participate and collaborate with other peers in the group. Apart from my mentors, I am specially happy to have worked with He Zhao from who I learnt the nuts and bolts of sampling in Bayesian models. To all of you, thank you for the insightful discussions, support and the human touch. I feel extremely honoured and privileged for this invaluable opportunity to grow as a scientist and human *down under*.

This dissertation would not have achieved the current status and scientific rigour without the comments and revision of many peers, from paper reviewers to academic committees. I feel very honoured to have received the thorough revision from Dr. Jerónimo Hernández and the invaluable feedback from Dr. David Carrera and Prof. Josep Sole during the pre-defence. Similarly, I am very pleased that Dr. Jose M. Peña and Dr. Mark Carman accepted to read this thesis, provide very constructive feedback and write the external reports. This thesis has also been greatly enriched from the co-supervision of three Master thesis. I feel very honoured to have supervised Gonzalo Pericacho, Carlos Cortes and Alberto Pou, not only for all the technical things I have learnt from them, but also for their determination to excel in their thesis.

Over the past few years, I also had the chance to engage in scientific discussions with many researchers, students and colleagues who have supported me in many different ways. Many thanks to all of you and, specially to Dr. Jordi Nin with who we wrote the fellowship application and who mentored me during my first year. He also connected me with Jose Cordero, Jordi Aranda and David Solans from who I have learnt many *hacking* tips and

with who we participated in my first *datathon*. Likewise, I am extremely grateful to Dr. Aleix Ruiz de Villa for bringing mathematical rigour in our discussions and for inviting me to join the local community of machine learning researchers and professionals. This community has broaden my knowledge in machine learning and expanded my professional network. Finally, I keep a vivid memory of my lab mates at the autonomic systems research group during the last months at the university. Special thanks to my running mate Fabrizio Pistagna, Dr. Cesare Cugnasco and Ferran Galí for the co-leadership of the Apache Spark meetup and to Miriam Bellver and Victor Campos for the Deep Learning sessions.

Finally, this journey would not have been possible without the encouragement and love of my family. I feel immense gratitude for my mum and dad, who instilled grit ever since I can remember and for my brothers, who have showed me the way. I am also very thankful to my sister-in-law and nephews for their joy and sweetness, to my uncle for his support and to Isabel for taking care of our family. Last but not least, I am and will be eternally grateful to Emma, my friend, partner and love, for being by my side in the good and bad moments and for always encouraging me to find purpose in every step.

# Abstract

Recent technological advances have radically changed the way we communicate. Today’s communication has become ubiquitous and it has fostered the need for information that is easier to create, spread and consume. As a consequence, we have experienced the shortening of text messages in mediums ranging from electronic mailing, instant messaging to microblogging. Moreover, the ubiquity and fast-paced nature of these mediums have promoted their use for previously unimaginable tasks. For instance, reporting real-world events was classically carried out by news reporters, but, nowadays, most interesting events are first disclosed on social networks like Twitter by eyewitness through short text messages. As a result, the exploitation of the thematic content in short text has captured the interest of both research and industry.

Topic models are a type of probability models that have traditionally been used to explore this thematic content, a.k.a. topics, in regular text. Most popular topic models fall into the sub-class of LVMs (Latent Variable Models), which include several latent variables at the corpus, document and word levels to summarise the topics at each level. However, classical LVM-based topic models struggle to learn semantically meaningful topics in short text because the lack of co-occurring words within a document hampers the estimation of the local latent variables at the document level. To overcome this limitation, pooling and hierarchical Bayesian strategies that leverage on contextual information have been essential to improve the quality of topics in short text.

In this thesis, we study the problem of learning semantically meaningful and predictive representations of text in two distinct phases:

- In the first phase, Part I, we investigate the use of LVM-based topic models for the specific task of event detection in Twitter. In this situation, the use of contextual information to pool tweets together comes naturally. Thus, we first extend an existing clustering algorithm for event detection to use the topics learned from pooled tweets. Then, we propose a probability model that integrates topic modelling and clustering to enable the flow of information between both components.
- In the second phase, Part II and Part III, we challenge the use of local latent variables in LVMs, specifically when the context of short messages is not available. First of all, we study the evaluation of the generalization capabilities of LVMs like PFA (Poisson Factor Analysis) and propose unbiased estimation methods to approximate it. With the most accurate method, we compare the generalization of chordal models without latent variables to that of PFA topic models in short and regular text collections.

In summary, we demonstrate that by integrating clustering and topic modelling, the performance of event detection techniques in Twitter is improved due to the interaction between both components. Moreover, we develop several unbiased likelihood estimation methods for assessing the generalization of PFA and we empirically validate their accuracy in different document collections. Finally, we show that we can learn chordal models without latent variables in text through *Chordalysis*, and that they can be a competitive alternative to classical topic models, particularly in short text.





# Contents

|  |             |
|--|-------------|
| <b>Acknowledgements</b>  | <b>v</b>    |
| <b>Abstract</b>  | <b>vii</b>  |
| <b>List of Figures</b>   | <b>xi</b>   |
| <b>List of Tables</b>  | <b>xiii</b> |
| <b>1 Introduction</b>  | <b>1</b>    |
| 1.1 Research Questions . . . . .                               | 4           |
| 1.2 Methodology . . . . .                                      | 6           |
| 1.3 Major Contributions . . . . .                              | 7           |
| 1.4 A Note on the Notation . . . . .                           | 8           |
| 1.5 List of Published Papers . . . . .                         | 8           |
| 1.6 Dissertation Outline . . . . .                             | 10          |
| <b>2 Probability Models for Text</b>                           | <b>11</b>   |
| 2.1 The Exponential Family . . . . .                           | 12          |
| 2.2 Probabilistic Graphical Models (PGMs) . . . . .            | 13          |
| 2.3 Text Representation . . . . .                              | 17          |
| 2.4 Topic Models . . . . .                                     | 19          |
| 2.5 Bayesian Inference . . . . .                               | 31          |
| 2.6 Evaluation . . . . .                                       | 35          |
| <b>I Event Detection Task</b>                                  | <b>39</b>   |
| <b>3 Event Detection in Twitter</b>                            | <b>41</b>   |
| 3.1 Event Detection: Problem Definition . . . . .              | 43          |
| 3.2 Related Work . . . . .                                     | 45          |
| 3.3 “La Mercè”: a Data Set for Local Event Detection . . . . . | 48          |
| 3.4 Summary and Conclusion . . . . .                           | 51          |
| <b>4 Tweet-SCAN: a Heuristic Approach</b>                      | <b>53</b>   |
| 4.1 Tweet-SCAN: a Heuristic Algorithm . . . . .                | 54          |
| 4.2 Experimentation . . . . .                                  | 59          |
| 4.3 Summary and Conclusion . . . . .                           | 64          |
| <b>5 WARBLE: a Probabilistic Approach</b>                      | <b>67</b>   |
| 5.1 WARBLE: the Probability Model . . . . .                    | 69          |
| 5.2 Learning Scheme . . . . .                                  | 74          |

|            |   |            |
|------------|---|------------|
| 5.3        | Experimentation . . . . .                               | 78         |
| 5.4        | Summary and Conclusion . . . . .                        | 84         |
| <b>II</b>  | <b>Likelihood Evaluation</b>                            | <b>85</b>  |
| <b>6</b>   | <b>Likelihood Estimation in Poisson Factor Analysis</b> | <b>87</b>  |
| 6.1        | Problem Definition . . . . .                            | 89         |
| 6.2        | Related Work . . . . .                                  | 93         |
| 6.3        | Experimental Setup . . . . .                            | 96         |
| 6.4        | Summary and Conclusion . . . . .                        | 97         |
| <b>7</b>   | <b>Left-to-right Sequential Samplers</b>                | <b>99</b>  |
| 7.1        | A Left-to-right Sampler for PFA . . . . .               | 100        |
| 7.2        | A Left-to-right Sampler for BPFA . . . . .              | 105        |
| 7.3        | Empirical Evaluation . . . . .                          | 107        |
| 7.4        | Summary and Conclusion . . . . .                        | 112        |
| <b>8</b>   | <b>Mean-field Variational Importance Sampling</b>       | <b>113</b> |
| 8.1        | Mean-Field Variational Importance Sampling . . . . .    | 114        |
| 8.2        | Mean-field VIS for PFA . . . . .                        | 117        |
| 8.3        | Mean-field VIS for BPFA . . . . .                       | 124        |
| 8.4        | Experimentation . . . . .                               | 130        |
| 8.5        | Summary and Conclusion . . . . .                        | 135        |
| <b>III</b> | <b>Chordal Models</b>                                   | <b>137</b> |
| <b>9</b>   | <b>Learning Chordal Models on Binarised Text</b>        | <b>139</b> |
| 9.1        | The <i>Chordalysis</i> algorithm . . . . .              | 140        |
| 9.2        | Related Work . . . . .                                  | 146        |
| 9.3        | Experimental Methodology . . . . .                      | 148        |
| 9.4        | Experiment Results . . . . .                            | 151        |
| 9.5        | Conclusion . . . . .                                    | 161        |
| <b>10</b>  | <b>Future Work and Conclusion</b>                       | <b>163</b> |
| 10.1       | Future Work . . . . .                                   | 163        |
| 10.2       | Conclusion . . . . .                                    | 166        |
|            | <b>Bibliography</b>                                     | <b>171</b> |
|            | <b>Acronyms</b>   | <b>183</b> |
|            | <b>Appendices</b>                                       | <b>189</b> |

# List of Figures

|      |  |    |
|------|--|----|
| 1.1  | Probabilistic Topic Model (Blei, 2012).  | 2  |
| 1.2  | Topic model with tweets pooled by user.  | 3  |
| 1.3  | Chordal Graphical Model (CGM).   | 5  |
| 1.4  | Box’s loop (Blei, 2014).   | 6  |
| 2.1  | Classification of probability models (Murphy, 2012).   | 13 |
| 2.2  | Latent Variable Model (LVM).   | 14 |
| 2.3  | Chordal Graphical Model (CGM).   | 16 |
| 2.4  | Bag-of-words or Unigram model.   | 17 |
| 2.5  | Mixture of Unigrams (MoU) graphical models.  | 20 |
| 2.6  | LDA (Latent Dirichlet Allocation) graphical model.   | 21 |
| 2.7  | HDP graphical model.   | 23 |
| 2.8  | mPCA (Multinomial Principal Component Analysis) graphical model.   | 24 |
| 2.9  | GaP (Gamma Poisson) graphical model (I).   | 25 |
| 2.10 | GaP graphical model (II).  | 25 |
| 2.11 | GaP graphical model (III).   | 27 |
| 2.12 | $\beta\gamma\Gamma$ -PFA graphical model.  | 28 |
| 2.13 | Latent Tree Models (LTMs).   | 29 |
| 2.14 | Restricted Boltzmann Machines (RBMs).  | 30 |
| 3.1  | Tweets generated during “La Mercè” 2014.   | 50 |
| 4.1  | DBSCAN (Density-based Spatial Clustering of Applications with Noise) definitions.                            | 55 |
| 4.2  | Text model scheme.   | 58 |
| 4.3  | Set matching metrics as a function of $\epsilon_3$ .   | 59 |
| 4.4  | Spatial representation of “wine tasting” and “food market” events.   | 61 |
| 4.5  | Event discrimination in “La Mercè” 2015 when pooling tweets by hashtag.                                      | 62 |
| 4.6  | Event discrimination in “La Mercè” 2015 when pooling tweets by keyword.                                      | 62 |
| 4.7  | Tweet-SCAN for different $\mu$ values.   | 63 |
| 4.8  | Tweet-SCAN for different $\epsilon_1, \epsilon_2, \epsilon_3$ values.  | 64 |
| 5.1  | Simplified PGMs (Probabilistic Graphical Models).  | 69 |
| 5.2  | Temporal histogram distribution $1d\text{-Hist}(\cdot)$ .  | 71 |
| 5.3  | WARBLE topic model (TM).   | 71 |
| 5.4  | The WARBLE model in detail.  | 73 |
| 5.5  | Spatio-temporal backgrounds.   | 79 |
| 5.6  | Topic proportions ( $\theta'_{k:}/\sum_t \theta'_{kt}$ ) per component (5 detected events and 1 background). | 81 |
| 5.7  | Best-performing Tweet-SCAN configuration.  | 82 |
| 5.8  | F-measure detection performance.   | 83 |

|     |  |     |
|-----|--|-----|
| 5.9 | Visual comparison of results. . . . .  | 83  |
| 6.1 | PFA graphical models with global point estimates. . . . .                        | 89  |
| 6.2 | BPFA (Bernoulli PFA) graphical models with global point estimates. . . . .       | 90  |
| 7.1 | KL divergence between estimated and exact values. . . . .                        | 109 |
| 7.2 | KL divergence ratios of L2R and DS estimates. . . . .                            | 110 |
| 7.3 | Document log-likelihood as a function of the number of samples in PFA. . .       | 111 |
| 7.4 | Document log-likelihood as a function of the number of samples in BPFA. .        | 111 |
| 8.1 | Reverse vs. forward KL divergence. . . . .                                       | 116 |
| 8.2 | KL divergence between estimated and exact values. . . . .                        | 131 |
| 8.3 | Document log-likelihood as a function of the number of samples in PFA. . .       | 132 |
| 8.4 | Sandwiched estimates as a function of the number of samples in PFA. . . .        | 133 |
| 8.5 | Document log-likelihood as a function of the number of samples in BPFA. .        | 134 |
| 8.6 | Sandwiched estimates as a function of the number of samples in BPFA. . . .       | 134 |
| 9.1 | Examples of the three models for a single document. . . . .                      | 146 |
| 9.2 | Averaged log-likelihood and number of parameters as function of training size.   | 153 |
| 9.3 | Averaged log-likelihood as function of training size and of clique size. . . .   | 154 |
| 9.4 | Averaged log-likelihood as a function of training size. . . . .                  | 154 |
| 9.5 | Averaged log-likelihood as a function of training size in big collections. . . . | 155 |
| 9.6 | AUC-PR and RMSE comparison for CGMs. . . . .                                     | 156 |
| 9.7 | AUC-PR and RMSE comparison for Chord-qNML, BPFA and HLTA. . . . .                | 156 |
| 9.8 | Running times comparison. . . . .  | 160 |

# List of Tables

|     |  |     |
|-----|--|-----|
| 2.1 | Summary of existing Topic Models for text. . . . .                               | 19  |
| 3.1 | “La Mercè” local festivities data sets. . . . .                                  | 49  |
| 3.2 | Labelled events in “La Mercè”. . . . .   | 51  |
| 4.1 | F-measure per event in “La Mercè” . . . . .                                      | 60  |
| 5.1 | Functional forms for $q(X)$ . . . . .  | 77  |
| 5.2 | Hyperparameter settings. . . . .   | 79  |
| 5.3 | Recall figures and spatio-temporal features per event. . . . .                   | 80  |
| 5.4 | 5 most probable words per topic $\phi'_{t:}/\sum_v \phi'_{tv}$ . . . . .         | 81  |
| 6.1 | Summary of existing likelihood estimation methods for LDA, PFA and BPFA. . . . . | 93  |
| 6.2 | Document collections. . . . .  | 97  |
| 7.1 | Hyperparameters and configuration. . . . .                                       | 109 |
| 9.1 | Model hyper-parameters and algorithm parameters. . . . .                         | 152 |
| 9.2 | AUC-PR figures for the log-likelihood anomaly score in WS. . . . .               | 157 |
| 9.3 | AUC-PR figures for the log-likelihood anomaly score in 20Newsgroups. . . . .     | 158 |
| 9.4 | AUC-PR figures for the IDF-weighted anomaly score in WS. . . . .                 | 159 |
| 9.5 | AUC-PR figures for the IDF-weighted anomaly score in 20Newsgroups. . . . .       | 159 |
| B.1 | Basic probability distributions in Exponential family form. . . . .              | 196 |



# 1

## Introduction

*“The medium is the message”*

Marshall McLuhan, 1964

Written communication has undergone a profound transformation in recent years. The rapid adoption of telecommunication and information technologies together with the increase of worldwide literacy rates have brought into scene new mediums and more players. Unrecognisable compared with those initial efforts of communication in the form of cave paintings or even the later development of papyrus and writing systems which made it mobile. Today’s communication is mostly ubiquitous due to the existence of electronic, interconnected and portable devices.

The ubiquity of digital technology has given birth to new mediums that complement or replace existing ones. For instance, many day-to-day business activities are currently managed through the exchange of e-mails, informal conversations are kept uninterrupted in instant messaging applications, on-line newspapers are continuously refreshed and political marketing campaigns take advantage of social networks to influence voters through personalised messaging. Writing has also evolved accordingly to accommodate the new capabilities of these mediums. For example, text nowadays incorporates emojis to express feelings, passages are hyper-linked to external websites with related content and words can be explored to find similar publications.

The growth in the number of literate people, who are able to create, store and consume text, has also had a huge impact on the current deluge of data. With more people connected to Internet (more than 50% of the worldwide population in 2018), the number of e-mails, tweets, Wikipedia articles and other written material keeps increasing yearly. Therefore, all this textual information needs to be organized in some structured way in order to make it findable. Search engines have enabled the massive indexation of documents, i.e. web pages, articles, items, etc., as per their relevance to certain terms and they have played a key role to answer specific user queries.

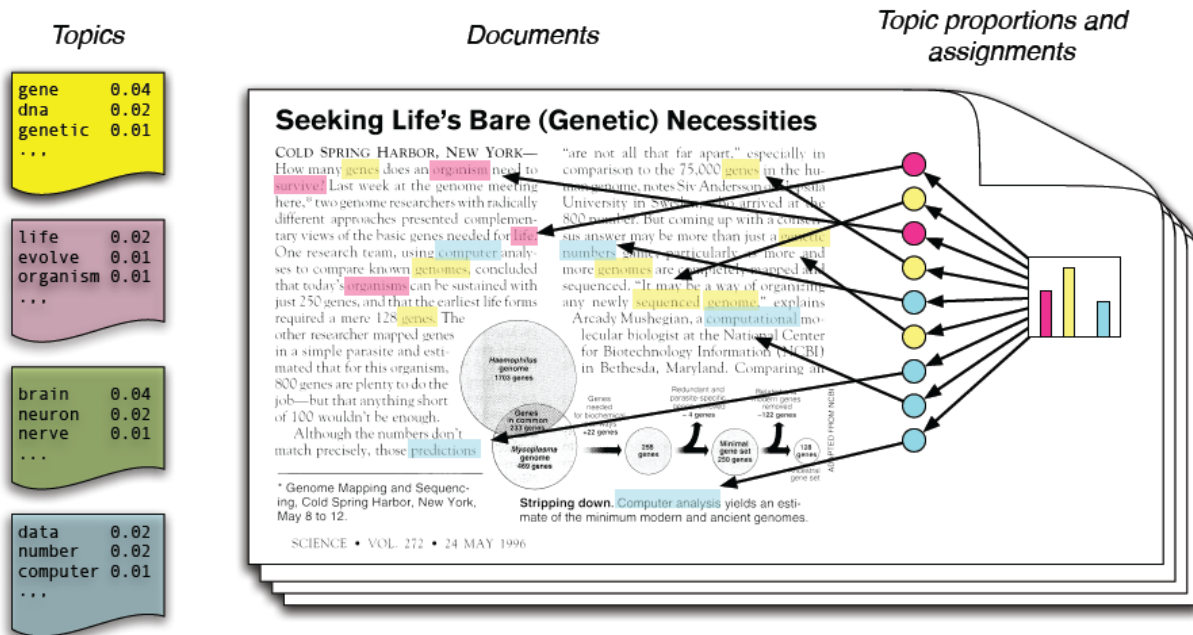


Figure 1.1: Probabilistic Topic Model (Blei, 2012).

However, search engines are not helpful to explore a collection of documents when the query is not clear. Imagine that an analyst receives a collection of documents whose content is unknown. First of all, the analyst will need to explore the themes of these documents in order to get an idea of what they are about. If the collection size is small, the exploration could be done by manually inspecting each document. For slightly larger collections, some statistics about the most relevant words could be insightful to the analyst to infer the content. As the collection grows, these solutions become impractical because not only the number of documents increases, but also does the vocabulary. In contrast, unsupervised machine learning methods excel at uncovering hidden patterns in growing data sets.

Probabilistic topic models (Blei, 2012), in particular, are unsupervised machine learning methods suitable for this thematic exploration of text. These models define a probability distribution over the set of documents which is then fitted to data in order to achieve good generalisation properties to unseen documents. The probability distribution contains several latent variables associated with words, documents and corpus. The word-level latent variables assign the corresponding topic to every word, whereas document and corpus latent variables provide useful summaries of topics at these levels.

In Fig. 1.1, the coloured nodes in the right hand side correspond to the word-level topic assignments. Each colour represents a distinct topic and topics are defined at the corpus level as the different probability distributions across the vocabulary. For instance, the yellow topic in Fig. 1.1 defines a probability distribution across the vocabulary with the probabilities set as in the table on the left hand side. The fact that the most likely words are “gene”, “DNA” and “genetic”, suggests that this topic might be related to genomics. By setting a number of topics to be much lower than the vocabulary size, these models act as dimensionality reduction methods in which documents can be represented in terms of topics. Latent variables at the document-level, like the probability distribution in the right hand-side of Fig. 1.1, summarise the proportion of topics in each document. For instance,



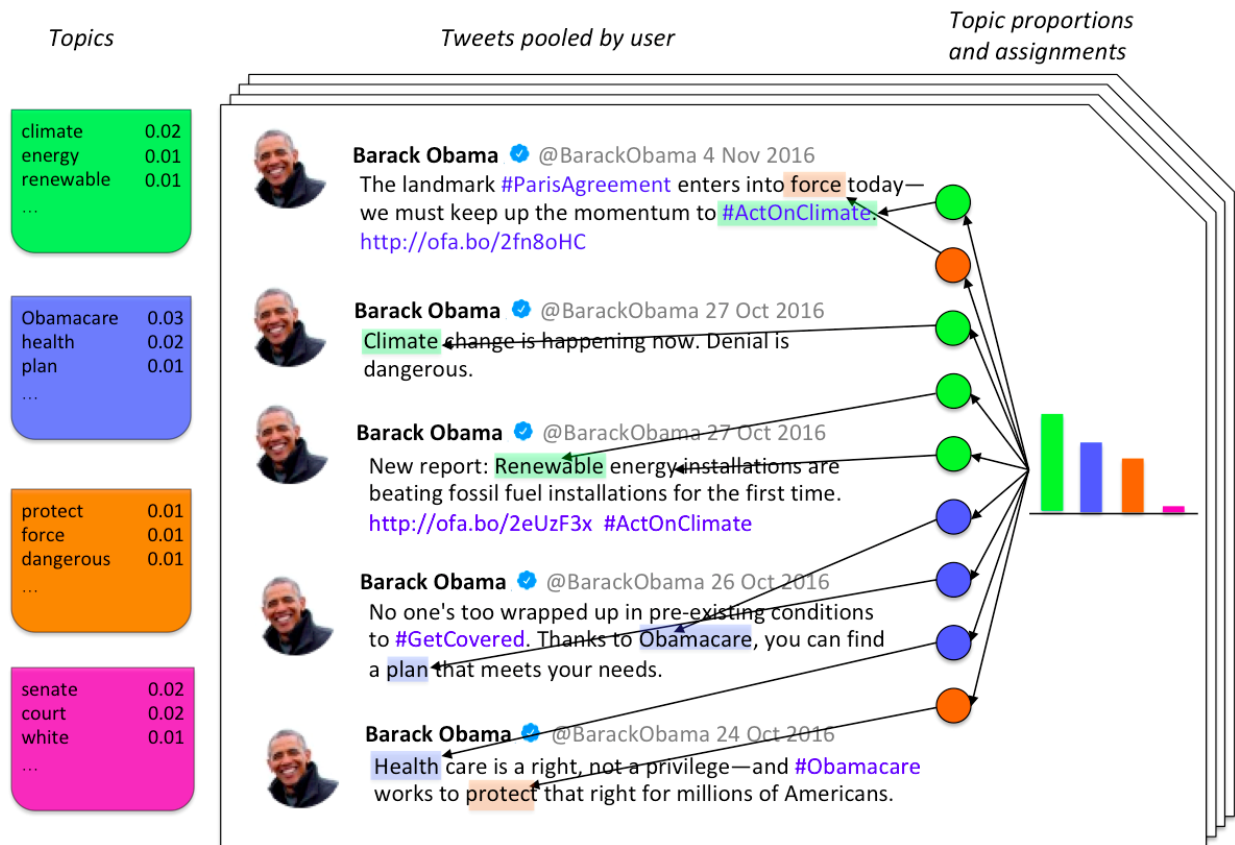


Figure 1.2: Topic model with tweets pooled by user.

according to Fig. 1.1, the document is a mix of three topics: genomics, evolution theory and computer science, but not neuroscience.

To exemplify how topic models work and point out their Achilles' heel, we make use of their generative story which shares some similarities to the writing process. The writing process starts with a writer deciding to communicate an idea via text. Assume that this idea can be described in terms of a mix of topics or themes. The number of topics ultimately depends on the interdisciplinarity of the field and the writer's style, but this should not be too high for obvious reasons of semantic coherence. In each topic, the writer has a subset of highly probable words that can be used to compose the text, i.e. in the genomics topic above, this subset of words are the yellow table in Fig. 1.1. The longer the document is, the more words from each topic are used and the higher the co-occurrence of words from a topic is. This co-occurrence of words is what defines the existence of semantic structure and the lack of it hampers to learn semantically meaningful and predictive topics. As suggested, the document length clearly impacts on the word co-occurrence, with very few co-occurrences in short text like tweets, text messages and headlines, among others. Therefore, classical topic models like the popular LDA (Latent Dirichlet Allocation) (Blei et al., 2003), which belongs to the class of LVMs (Latent Variable Models), are known to perform poorly with short text. Intuitively, these models have the same troubles than an unfamiliar reader trying to figure out what a 140-character-long tweet is about.

New digital mediums like Twitter are thought for fast creation and consumption of

information. This explains the limitation of the tweet length initially to 140 characters and currently to 280. In such social networks, users create their own sphere by following other users with whom they have overlapping interests. This determines the context of the communication in which these short text messages are exchanged. When readers know about the context, tweets also become more meaningful to them. Thus, it seems logical for topic models to leverage on contextual information to learn more meaningful thematic representations for short text. [Hong and Davison \(2010\)](#) studied different tweet pooling strategies in which context was built by putting tweets together and showed that not only topic coherence improves but also does the performance in external tasks for which topic models can be used. For instance, the pooling scheme in Fig. 1.2 aggregates tweets per user to mitigate the lack of co-occurrences in a single tweet. More principled approaches have been proposed to incorporate context to topic models through hierarchical structures and they have been shown to outperform classical pooling strategies ([Lim et al., 2016](#)). These approaches enable sharing of statistical strength by stacking up probability distributions that account for the context of the communication, i.e. authors, hash tags, network, etc..

## 1.1 Research Questions

Against this background, we wonder whether there exist specific tasks or queries on text for which the aggregation or the use of contextual information becomes more natural and effective. That is the case for the event detection task as per its definition in the TDT (Topic Detection and Tracking) project ([Allan et al., 1998](#)): “The notion of an event differs from a broader category of events both in spatial/temporal localization and in specificity. For example, the eruption of Mount Pinatubo on June 15th, 1991 is considered to be an event, whereas volcanic eruption in general, a class of events”. Under this definition of event, classes of events could conform to the idea of topics, while specific events could correspond to instantiations of these topics with a specific spatial/temporal context. Besides, in this setup we are not particularly interested in the individual documents but in the groupings of them that compose these events. Therefore, the first research question that we address goes as follows: **Can probability models that leverage on contextual information be effective for detecting events in mediums like Twitter?** To answer this question, we will take a pragmatic approach. Firstly, we will propose and study the detection performance of a heuristic algorithm that combines LVM-based topic models and event clustering via the pooling strategies in [Hong and Davison \(2010\)](#). Secondly, we will develop a LVM that jointly learns topics and events by automatically pooling tweets as per their spatial/temporal context. To conduct this experimental work, we will also introduce a data set for event detection in Twitter in which specific events are manually identified by domain experts. The evaluation of both methods will be conducted in terms of task-specific metrics, like precision and recall.

Nonetheless, the evaluation of probability models is commonly performed independently of the task at hand in terms of the probability of the unseen data given the model, which measures the generalisation capabilities of the model to unseen data. For most interesting LVM-based topic models, this probability is intractable to compute since it usually involves a sum or integral over a huge space of probabilities. The unbiased estimation of this probability for the well-known LDA model has attracted lot of interest and different estimation methods have been proposed ([Wallach et al., 2009b](#); [Buntine, 2009](#)). However, the same

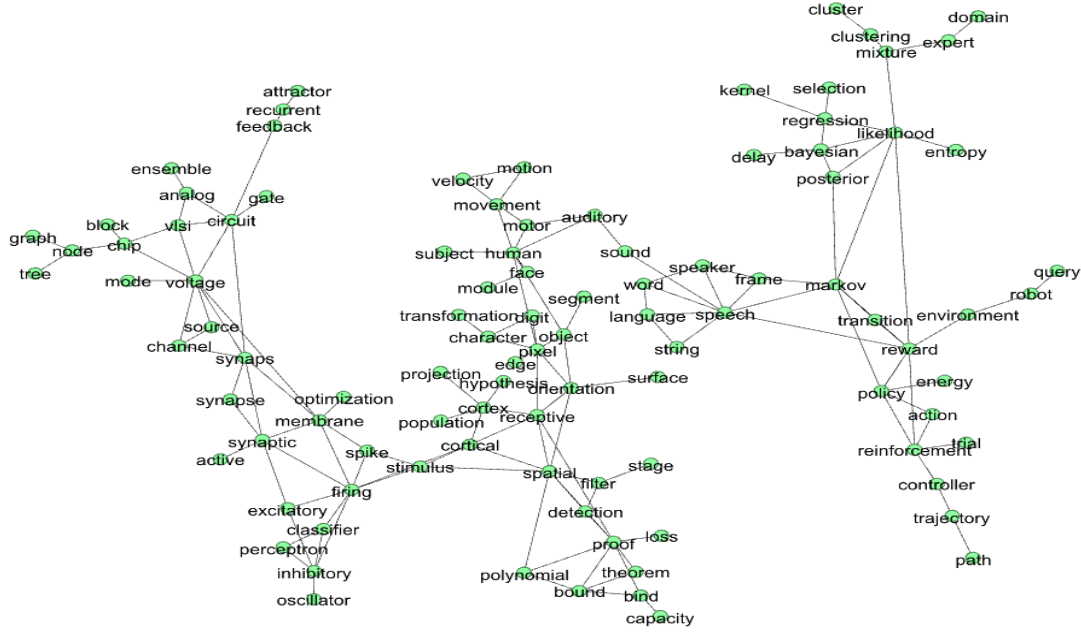


Figure 1.3: Chordal Graphical Model (CGM).

problem for a broader class of LVM-based topic models, referred here to as PFA (Poisson Factor Analysis) (Zhou et al., 2012), remains unexplored. PFA-like topic models are more appropriate than LDA-like ones for comparing across probability models that are based on the bagged representation of text and this study will pave the way to compare different types of probability models for text in a consistent way. As a result, the second research question can be formulated as follows: **Can we develop accurate likelihood estimation methods for PFA topic models?** To do this, we first extend the state-of-the-art Left-to-right sequential sampler proposed for LDA by Buntine (2009) to PFA and we then propose a different approach called VIS (Variational Importance Sampling) which gives rise to two distinct estimation methods. These methods will also be applied to the BPFA (Bernoulli PFA) model (Zhou, 2015), where the observed variables are binarised.

As discussed above, LVM-based topic models experience difficulties to learn meaningful topic representations in short text when contextual information is not available. A plausible explanation for this shortcoming is attributed to the fact that document and word level latent variables are less certain in short documents due to the lack of co-occurring words. See in Fig. 1.2 how challenging it is to figure out that the fourth tweet is about “health care” if one does not have more information about “Obamacare”. Therefore, probability models without these local variables could be, in principle, a better alternative to LVM-based models. This will bring us to the third research question of this thesis: **Can probability models without latent variables but with richer structures be good alternatives for text prediction?** To address this question, we will search for probability models within the subclass of CGMs (Chordal Graphical Models), which is an expressive class of graphical models that can be efficiently explored in data sets with thousands of variables through the *Chordalysis* algorithm (Petitjean and Webb, 2015a). The resulting models

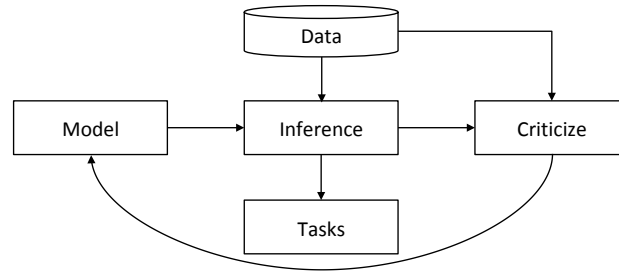


Figure 1.4: Box’s loop (Blei, 2014).

express the relationships (edges) that exist among words (nodes) through chordal graphs. For example, Fig. 1.3 plots the resulting chordal model learned from a collection of NIPS papers which have been processed with a vocabulary of 100 words. We can observe that words tend to group together in the graph as per their semantic meaning, so that words like “cluster”, “clustering” and “mixture” appear interconnected in the north-east part of the graph and they are far from words like “oscillator”, “inhibitory” and “perceptron”, which appear in the south-west. Therefore, we will use these chordal graphs to learn probability models for binarised text which have good generalisation in held-out data. To do that, we will propose new exploration metrics and parameter estimation methods for *Chordalysis* that are more appropriate for prediction tasks. Finally, the experimental work will compare CGMs, other representatives without local latent variables to PFA-like topic models in several collections with good representatives of short and long text. The comparison will assess their generalisation capabilities in terms of probability of unseen documents through the estimation methods developed earlier, but also in terms of other task-specific metrics such as omni-directional prediction and anomaly detection.

In a nutshell, we study the use of probability models for thematic exploration of text, with special emphasis on short text. We first focus on the event detection task in Twitter in which tweet pooling and the use of side information are intrinsic to the task and beneficial for short text. Then, we address the evaluation of LVM-based topic models in terms of their probability on unseen data. This probability is often intractable to compute and hence we study and propose different unbiased estimation methods. Finally, we question the use of document-level latent variables in topic modelling, especially for short text, and propose to learn probability models without latent variables but with a richer set of dependencies among the observed, known as chordal graphical models.

## 1.2 Methodology

In this thesis, we follow the probabilistic modeling approach to machine learning (Bishop, 2013; Blei, 2014; Ghahramani, 2015). In this approach, machine learning solutions explicitly state the modeling assumptions in a compact language enabling to change and upgrade them in an iterative manner. Fig. 1.4 shows the iterative process presented in (Blei, 2014) as an adaptation of Box’s perspective (Box, 1976). Next, we review how this thesis approaches each of the stages of Fig. 1.4:

- **Data.** The thesis revolves around text data, with particular emphasis on short text. A text corpus or data set is a collection of documents, each composed of a sequence

of words. A text corpus might contain meta-data such as authors, date, location, etc. which might be relevant to characterize the context of the communication. In the first part of the thesis, we create a hand-crafted data set from Twitter for the problem of event detection. In the second and third parts, we use existing and publicly available data sets of text with good representatives of short and long text.

- **Tasks.** Machine learning solutions are formulated with a particular (or multiple) task(s) in mind. We address the task of retrospective event detection from short text tweets as part of the more general task of thematic exploration of text.
- **Model.** The model refers to the mathematical formalisms used to express the assumptions of the data generation process. In this dissertation, we focus on the probability models whose conditional independences can be expressed in a graph, also known as PGMs (Probabilistic Graphical Models) (Koller and Friedman, 2009b). In particular, most probabilistic topic models are based on a particular type of PGM with latent variables known as LVM. Therefore, we will study this particular type of PGM, but we will also propose an alternative without latent variables and based on chordal graphs.
- **Inference.** This is the process to draw conclusions from the data. When performed in combination to probability models, statistical inference enable us to deal with uncertainty in a principled way. However, exact inference is intractable for most interesting models due to the coupling of multiple variables and their complex relationships. In this thesis, we will mainly use variational methods, but also Gibbs sampling, to approximate intractable inference. Although probabilistic programming tools (BUGS (Lunn et al., 2012), Edward (Tran et al., 2016), etc.) exist to automate this inference, we implement our own inference algorithm since these tools mostly work for continuous latent variables and do not always support every probability distribution.
- **Criticise.** This stage questions the model as well as the inference method against true data in order to confirm or refute their validity. In case the model is refuted, the feedback loop enables to change or upgrade some of the hypothesis in the model. Here, we will use the probability of unseen data to evaluate how well or poorly the joint solution (model+inference) performs. Other task-specific measures, like detection accuracy, have been considered to drive this iterative process.

## 1.3 Major Contributions

The technical contributions of this dissertation can be listed under each of the three research questions presented earlier.

**I. Can probability models that leverage on additional information be effective for detecting events in new mediums like Twitter?**

- We build a data set of public tweets for the task of event detection, in which events were manually tagged by domain experts.
- We extend DBSCAN (Density-based Spatial Clustering of Applications with Noise) to deal with textual features to uncover events in the previous data set.

- We propose a probability model and a learning algorithm for event detection from tweets.

## II. Can we develop accurate likelihood estimation methods for PFA topic models?

- We present L2R for PFA, a left-to-right sequential sampler initially proposed for LDA.
- We propose new estimation methods based on IS (Importance Sampling) with upper- and lower-bounded mean-field proposals.
- We tune up the proposed estimation methods for BPFA.

## III. Can probability models without latent variables but with richer structures be good alternatives for text prediction?

- We propose CGMs, an expressive class of graphical models without document-level latent variables, for thematic exploration of text.
- We study different metrics to explore the space of CGMs through *Chordalysis* and several smoothing techniques to improve the predictability on held-out text.
- We present a method to incorporate counts into CGMs learned from binarised text.

## 1.4 A Note on the Notation

In this thesis, we conduct applied and fundamental research in the field of probability models for text. To develop and present the research ideas in a formal way, we make use of the existing mathematical notation and terminology in the field. In Appendix A, we list the main conventions used throughout the thesis. We classify them with respect to the context that they are used to dispel confusion in case of the same term or notation is used to refer to different things, instead of changing notation. For instance,  $V$  is commonly used to refer to the number of vertices in graph theory and to the vocabulary size in text analysis. However, we also develop our own notation to distinguish between key aspects of this thesis. For example, we make explicit the distinction between the two common representations of text data. Sequence-specific notation represents words in a document using  $w$ ,  $\mathbf{w}_n$ ,  $\mathbf{w}_{nm}$  and  $\mathbf{W}$ , whereas we use  $y$ ,  $\mathbf{y}_n$ ,  $\mathbf{y}_{np}$  and  $\mathbf{Y}$  to refer to the corresponding concepts in the bagged representations.

## 1.5 List of Published Papers

Next, we present the list of research papers that have been written and published during the doctoral studies. We link the papers to the corresponding research question addressed in this thesis or to other research activities done in parallel.

- I Can probability models that leverage on contextual information be effective for detecting events in mediums like Twitter?



- Capdevila, J., Cerquides, J., Nin, J., and Torres, J. (2015). Tweet-SCAN: An event discovery technique for geo-located tweets. In *Artificial Intelligence Research and Development: Proceedings of the 18th International Conference of the Catalan Association for Artificial Intelligence*, volume 277, page 110. IOS Press
- Capdevila, J., Cerquides, J., and Torres, J. (2016b). Recognizing warblers: a probabilistic model for event detection in twitter. Presented at the Workshop of Anomaly Detection at the International Conference on Machine Learning (ICML)
- Capdevila, J., Cerquides, J., Nin, J., and Torres, J. (2017a). Tweet-SCAN: An event discovery technique for geo-located tweets. *Pattern Recognition Letters*, 93:58 – 68. Pattern Recognition Techniques in Data Mining
- Capdevila, J., Cerquides, J., and Torres, J. (2017b). Event detection in location-based social networks. In *Data Science and Big Data: An Environment of Computational Intelligence*, pages 161–186. Springer
- Capdevila, J., Cerquides, J., and Torres, J. (2018a). Mining urban events from the tweet stream through a probabilistic mixture model. *Data Mining and Knowledge Discovery*, 32(3):764–786

## II Can we develop accurate likelihood estimation methods for PFA topic models?

- Capdevila, J., Cerquides, J., Torres, J., Petitjean, F., and Buntine, W. (2018c). A left-to-right algorithm for likelihood estimation in gamma-poisson factor analysis. In *Joint European Conference on Machine Learning and Knowledge Discovery in Databases*, pages 638–654. Springer

## III Can probability models without latent variables but with richer structures be good alternatives for text prediction?

- Capdevila, J., Zhao, H., Petitjean, F., and Buntine, W. (2018d). Experiments with learning graphical models on text. *Behaviormetrika* <https://doi.org/10.1007/s41237-018-0050-3>

## IV As a product of other research activities:

- [Joan Capdevila’s MSc Thesis] Capdevila, J., Arias, M., and Arratia, A. (2016a). GeoSRS: A hybrid social recommender system for geolocated data. *Information Systems*, 57:111 – 128
- [Gonzalo Pericacho’s MSc Thesis] Capdevila, J., Pericacho, G., Torres, J., and Cerquides, J. (2016c). Scaling DBSCAN-like algorithms for event detection systems in twitter. In *International Conference on Algorithms and Architectures for Parallel Processing*, volume 10048, pages 356–373. Springer.
- [5th BSC Severo Ochoa Doctoral Symposium] Capdevila, J., Cerquides, J., and Torres, J. (2018b). Model-based machine learning for retrospective event detection. Presented at 5th BSC Severo Ochoa Doctoral Symposium

## 1.6 Dissertation Outline

The rest of this dissertation is split into three parts preceded by the preliminary work in Chapter 2, which contains the basics of probability models for text. Each part is self-contained with its own problem statement and related work and it addresses one of the three research questions presented earlier.

Part I of this dissertation leverages on side information and pooling methods to carry out the task of event detection in Twitter. Chapter 3 presents the problem of event detection in Twitter, reviews the existing literature and describes a data set suitable for detection of local events. Then, Chapter 4 extends a well-known heuristic algorithm for clustering to deal with topics that are learned from pooled tweets. Chapter 5 describes WARBLE, a new probability model that jointly uncovers events and topics.

Part II studies the problem of assessing the document likelihood in PFA. Chapter 6 explains the problem of likelihood evaluation and present a methodology to evaluate estimation methods. Chapter 7 presents a left-to-right algorithm to decompose the document estimation problem into smaller sub-problems. Chapter 8 introduces a class of methods to perform IS with factorised distributions as proposal distributions and propose the use of variational bounds to sandwich the estimates of the document likelihood.

Part III proposes CGM as an alternative for text modelling. In particular, Chapter 9 introduces new scoring functions and parameter estimation methods for *Chordalysis* to learn expressive graphical models for binarised text. This chapter also present the experimental results of comparing a wide range of graphical models with and without latent variables in short and regular text.

Finally, Chapter 10 points at future research in each of the three research lines developed along the thesis and it summarises the main conclusions of this work.



# 2

## Probability Models for Text

*“What I cannot create, I do not understand”*

Richard FEYNMAN, 1988

This chapter introduces the basic building blocks for this thesis, the so-called probability models. We can think of probability or probabilistic models as the set of simplifying assumptions about the problem that we are presented with. Thus, the understanding of basic probability models is essential to use them for certain tasks and to build tailored models for the problem at hand.

For instance, a professional gambler who bets on the outcomes of a game often builds a model about the game. After having observed several games, the gambler has also acquired some knowledge about the most likely events. Based on both the model and the recorded games, the gambler bets on the most likely outcomes. Therefore, the success of the gambler ultimately depends on how well his/her simplified model represents the true game and how much data have been able to gather.

The generative process of text, i.e. the set of rules by which we write, is also unknown, and even more relevant, too complex to specify. Therefore, the probability models for text presented in this chapter will trade off complexity and utility. This means that we will make simplifying assumptions about the generation process of text in order to present models that have practical uses for thematic exploration of text.

The contents of this chapter are organized as follows. In Section 2.1, we introduce the exponential family, a wide group of probability distributions which embraces most of the discrete and continuous distributions used in this thesis. We then present in Section 2.2 an approach to build more complex probability models based on a compact graphical representation known as PGMs (Probabilistic Graphical Models). In Section 2.3, we introduce two common representations of text in the literature which only differ on a combinatoric term, but they give rise to two classes of probability models for text. After that in Section 2.4, we review different probability models for text whose conditional independences can be ex-

pressed in the graphical language introduced earlier. In Section 2.5, we present how these models can be learned from data and how certain tasks can be formulated as inferences. Finally, we will discuss the evaluation of these models in Section 2.6.

## 2.1 The Exponential Family

The Exponential family embraces several probability distributions that have a particular form with useful algebraic properties. Discrete members of this family are the Bernoulli, Categorical, Multinomial with also known number of trials, Poisson and Negative Multinomial with known number of failures. The Normal, Beta, Dirichlet, Gamma are some good representatives of the continuous distributions in this family.

The probability density or mass function for a continuous or discrete random variable  $\mathbf{x} = (x_1, \dots, x_d) \in \mathcal{X}^d$  in the exponential family can be written as,

$$p(\mathbf{x}; \boldsymbol{\theta}) = h(\mathbf{x}) \exp(\boldsymbol{\theta}^T t(\mathbf{x}) - A(\boldsymbol{\theta})) \quad (2.1)$$

where  $\boldsymbol{\theta} = (\theta_1, \dots, \theta_k) \in \Theta \subseteq \mathbb{R}^K$  are the natural or canonical parameters,  $t(\mathbf{x})$  is a vector of sufficient statistics,  $h(\mathbf{x})$  is the base measure and  $A(\boldsymbol{\theta})$  is the log-partition function or cumulant which normalises the distribution as follows,

$$A(\boldsymbol{\theta}) = \log \int h(\mathbf{x}) \exp(\boldsymbol{\theta}^T t(\mathbf{x})) d\mathbf{x}. \quad (2.2)$$

A more general form for the exponential family is given by,

$$p(\mathbf{x}; \boldsymbol{\theta}) = h(\mathbf{x}) \exp(\boldsymbol{\eta}(\boldsymbol{\theta})^T t(\mathbf{x}) - A(\boldsymbol{\eta}(\boldsymbol{\theta}))) \quad (2.3)$$

where  $\boldsymbol{\eta}()$  is the function that maps from parameter  $\boldsymbol{\theta}$  to the canonical parameters  $\boldsymbol{\eta} = \boldsymbol{\eta}(\boldsymbol{\theta})$ . For example, the Bernoulli distribution with binary events  $x \in \{0, 1\}$  is expressed in its canonical form through the natural parameter  $\boldsymbol{\eta}$ , the function mapping from parameters  $\boldsymbol{\eta} = \boldsymbol{\theta} = \{p\}$  to  $\boldsymbol{\eta}(\boldsymbol{\theta}) = \log \frac{p}{1-p}$  where  $p$  is the success probability of  $x = 1$ , a sufficient statistic  $t(x) = x$ , a base measure  $h(x) = 1$  and a log-partition function  $A(\boldsymbol{\eta}) = \log(1 + e^{\boldsymbol{\eta}})$ . In Appendix B.1, we present the common parametrisation of the probability distributions used in the thesis. Moreover, Table B.1 summarises their parametrisation in the exponential family.

An important property of the Exponential family for Bayesian statistics is that one can obtain a conjugate prior to a likelihood in the Exponential family by considering the following form for the prior over parameters  $\boldsymbol{\eta}$  in Eq. (2.3),

$$p(\boldsymbol{\eta}; \boldsymbol{\chi}) = h(\boldsymbol{\eta}) \exp(\boldsymbol{\chi}^T t(\boldsymbol{\eta}) - A(\boldsymbol{\chi})) \quad (2.4)$$

where  $\boldsymbol{\chi} = (\boldsymbol{\chi}_1, \chi_2)$  is a vector whose components  $\boldsymbol{\chi}_1$  is also a vector of the same size as  $\boldsymbol{\eta}$  and  $\chi_2$  is a scalar. Then, the sufficient statistics have to satisfy that  $t(\boldsymbol{\eta}) = (\boldsymbol{\eta}, -A(\boldsymbol{\eta}))$ . A prior distribution is conjugate to a likelihood, if the normalised product of both distributions gives a posterior distribution which is in the same form than the prior. For instance, a conjugate prior for the Bernoulli distribution above can be derived by setting up a distribution with natural parameters  $\boldsymbol{\chi} = (\boldsymbol{\chi}_1, \chi_2)$  and sufficient statistics  $t(\boldsymbol{\eta}) = (\boldsymbol{\eta}, -A(\boldsymbol{\eta})) = (\boldsymbol{\eta}, -\log(1+e^{\boldsymbol{\eta}}))$ . If we express the sufficient statistics w.r.t. parameters  $\boldsymbol{\theta} = p$ ,

then  $t(\theta) = (\log \frac{p}{1-p}, \log(1-p))$ , one observes that the Beta distribution parametrised as  $\text{Beta}(\alpha = \chi_1, \beta = \chi_2 - \chi_1)$  is a conjugate prior to the Bernoulli. Similarly, one can derive other well-known conjugacies for Binomial-Beta, Poisson-Gamma, Multinomial-Dirichlet and Categorical-Dirichlet.

## 2.2 Probabilistic Graphical Models (PGMs)

In this section, we introduce PGMs, a sub-class of probability models which allows a graphical representation of the conditional independences among the random variables. Fig. 2.1 sketches the different subclasses of probability models that exists. Within graphical models, there are two main sub-classes called Directed and Undirected, which differ on the type of conditional independences able to model through directed or undirected graphs. Chordal or decomposable models appear in the intersection of both classes, since they represent distributions that can be modelled by either directed or undirected graphs. In the following subsections, we review in detail the type of conditional independences represented by each class.

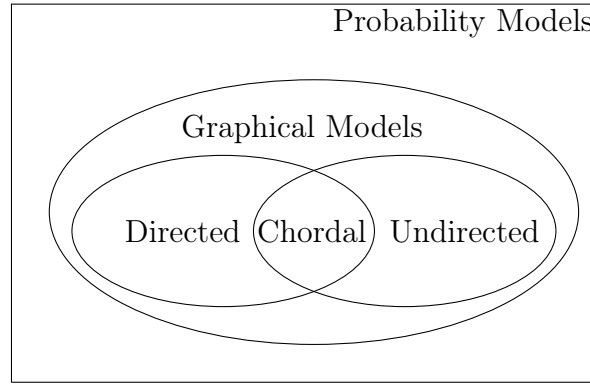


Figure 2.1: Classification of probability models (Murphy, 2012).

### 2.2.1 Directed Graphical Models (DGMs)

DGMs (Directed Graphical Models), also known as Bayesian, belief or causal networks, are a subclass of graphical models whose nodes in the graph are ordered following a topological ordering. Topological sorting is a property of directed graphs such that no parent occurs after their children. At least one topological ordering exists in any graph if and only if the graph has no directed cycles, that is to say, if it is a DAGs (Directed Acyclic Graphs).

In the following, we assume that a DAGs  $\mathcal{G}$  has  $V$  nodes ( $V \in \mathbb{N}$ ) and we use  $v$  ( $v \in \{1, \dots, V\}$ ) to denote a node label in  $\mathcal{G}$ . The type of conditional independences that a DGM encodes about its variables  $\mathbf{x} = \{x_1, \dots, x_V\}$  can be expressed in terms of the relationships among the corresponding nodes in  $\mathcal{G}$ . In particular, we have that a variable  $x_v$  is conditionally independent of all its predecessors given its parents. That is,

$$x_v \perp\!\!\!\perp \mathbf{x}_{\text{pred}(v)} \mid \mathbf{x}_{\text{par}(v)} \quad (2.5)$$

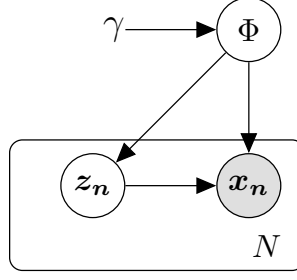


Figure 2.2: Latent Variable Model (LVM).

where  $\text{pred}(v)$  and  $\text{par}(v)$  represent the set of predecessors and parents of  $v$  in  $\mathcal{G}$ , respectively. This property enables the factorisation of the probability distribution from the graph as follows,

$$p(\mathbf{x}) = \prod_{v=1}^V p(x_v | \mathbf{x}_{\text{par}(v)}; \boldsymbol{\theta}_v). \quad (2.6)$$

where  $p(x_v | \mathbf{x}_{\text{par}(v)}; \boldsymbol{\theta}_v)$  is the conditional probability of variable  $x_v$  given its parents  $\mathbf{x}_{\text{par}(v)}$  and parametrised by  $\boldsymbol{\theta}_v$ .

An interesting subgroup of DGMs are the so-called template models (Koller and Friedman, 2009a). These are models that have some interesting recurrences in their variables that enable their specification through a more compact graphical representation. Next, we review LVMs (Latent Variable Models) (Blei, 2014) a type of template model that encompasses many of the probability models for text that we later review.

### 2.2.1.1 Latent Variable Models (LVMs)

LVMs (Blei, 2014) (Blei, 2014) defines a sub-class of DGMs with a specific recurrent relationship of the variables in the model. This sub-class embraces a broad range of probability models for text like mixture models, mixed-membership models and matrix factorisation, but also for other types of data and problems like linear factor models and time-series.

Fig. 2.2 shows a LVM graphical model in Plate notation. As the name suggests, LVMs contains latent or hidden variables ( $\mathbf{z}, \Phi$ ), white nodes in the graph, combined with observed variables ( $\mathbf{x}_n$ ), shaded nodes. Moreover, this template model contains two types of variables: local and global. Local variables  $\{\mathbf{z}_n, \mathbf{x}_n\}$  are defined at the observation level, whereas global variables  $\Phi$  are shared by all the observations. The plate grouping the local variables, which is indexed by the number of observations  $N$ , indicates that the local variables are repeated  $N$  times. Therefore, this model assumes that the  $n$ -th local variables  $\{\mathbf{z}_n, \mathbf{x}_n\}$  are independent from the rest of local variables  $\{\mathbf{z}_{\setminus n}, \mathbf{x}_{\setminus n}\}$  given the global variables  $\Phi$ . Moreover, the graphical representation also displays hyperparameters like  $\gamma$  and  $N$  in plain text attached to the variable or plate they refer to.

Furthermore, conditionally conjugate LVMs assume that the complete conditionals, i.e. the probabilities of each variable conditioned to all other variables in the model, are in the Exponential family. Because of this assumption, the conditionals of both local and observed variables as well as the prior on the global variables are also in the exponential family and they can often be set accordingly to obtain a conditionally conjugate model.

Mathematically, this means that these conditionals have to be of the form,

$$p(x_n, z_n | \phi) = h(x_n, z_n) \exp(\Phi^T t(w_n, z_n) - A_l(\Phi)) \quad (2.7)$$

$$p(\Phi; \gamma) = h(\Phi) \exp(\gamma^T t(\Phi) - A_g(\gamma)) \quad (2.8)$$

and the sufficient statistics of  $p(\Phi; \gamma)$  must be equal to  $t(\Phi, -A_l(\Phi))$ , as seen in section 2.1 for conjugate distributions. This type of conditional conjugacies between global and local variables are less strict than requiring full conjugacy. For example, a conditionally conjugate global prior for a GMM (Gaussian Mixture Model) would be independent Dirichlet and NIW (Normal-Inverse-Wishart) distributions.

### 2.2.2 Undirected Graphical Models (UGMs)

UGMs (Undirected Graphical Models), also called Markov Random Fields or simply Markov Networks, are a subclass of graphical models whose nodes are connected through undirected edges. This subclass of graphs is more appropriate to represent probability distributions in which the direction of the correlation is not clear, e.g. the interaction between neighbouring pixels in an image.

Next, we assume an undirected graph  $\mathcal{G}$  with  $S$  nodes whose node labels are depicted by  $s \in \{1, \dots, S\}$ . The conditional independences of an UGM with variables  $\mathbf{x} = \{x_1, \dots, x_S\}$  can be expressed in this graph  $\mathcal{G}$  as,

$$x_s \perp\!\!\!\perp \mathcal{X} \setminus \text{cl}(x_s) \mid \text{mb}(x_s) \quad (2.9)$$

where  $x_s$  is any of the variables in the UGM,  $\text{mb}(x_s)$  is the *Markov blanket* of  $x_s$  which corresponds to the immediate neighbours of  $x_s$  for undirected graphs and  $\text{cl}(x_s)$  is the *closure* of  $x_s$  defined as the union of the *Markov blanket* with itself,  $\text{mb}(x_s) \cup x_s$ . This means that  $x_s$  is conditionally independent of all other variables given its immediate neighbours.

The factorisation of the joint distribution in UGMs is not as straightforward as in DGM, since undirected graphs do not provide a natural way to factorise this distribution. Nonetheless, the Hammersley-Clifford theorem formulates the joint through *potential functions* or *factors* associated to each *maximal clique*. A *potential function* or *factor* is any non-negative function of its arguments. A *clique* is a subset of nodes in the graph such that the induced subgraph is complete, every pair of nodes are connected. Then, a *clique* is *maximal* if it cannot be enlarged without breaking the *clique* property. More precisely, the Hammersley-Clifford theorem says that any positive distribution whose conditional independences can be expressed through a UGM can be represented as,

$$p(\mathbf{x}; \boldsymbol{\theta}) = \frac{1}{Z(\boldsymbol{\theta})} \prod_{c \in \mathcal{C}} \psi_c(\mathbf{x}_c; \boldsymbol{\theta}_c) \quad (2.10)$$

where  $\mathcal{C}$  represent the set of maximal cliques,  $\psi_c()$  is the potential function corresponding to clique  $c$  and parametrised through  $\boldsymbol{\theta}_c$ .  $Z(\boldsymbol{\theta})$  is the *partition* function or normalising constant which is given by,

$$Z(\boldsymbol{\theta}) = \sum_{\mathbf{x}} \prod_{c \in \mathcal{C}} \psi_c(\mathbf{x}_c; \boldsymbol{\theta}_c) \quad (2.11)$$

where  $\sum_{\mathbf{x}}$  represents the sum over all possible states of  $\mathbf{x}$ . This sum or integral if  $\mathbf{x}$  is continuous is usually intractable because it involves too many states.

### 2.2.3 Chordal Graphical Models (CGMs)

CGMs (Chordal Graphical Models) or decomposable models are a subset of graphical models that can faithfully represent a probability distribution either through a DGM or an UGM. A graphical model is chordal or decomposable if all cycles in the induced graph that pass through more than 3 nodes contain a *chord*, that is an edge that connects two of the nodes in the cycle but it is not part of it.

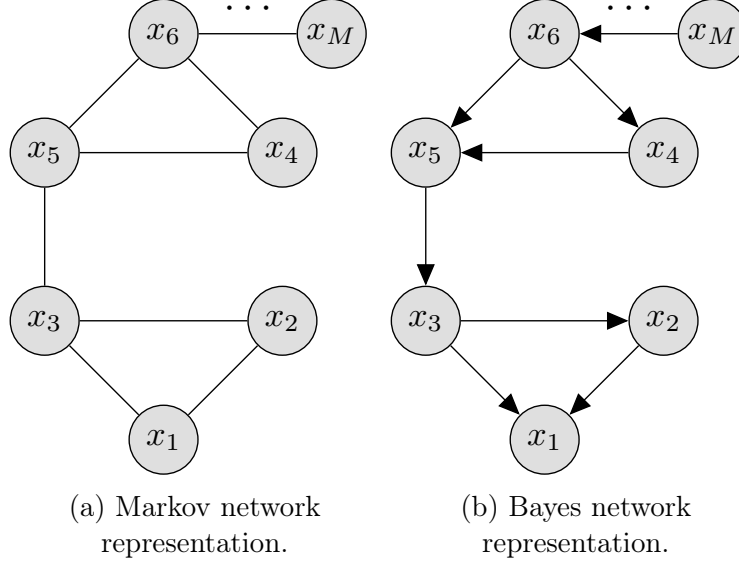


Figure 2.3: Chordal Graphical Model (CGM).

In Fig. 2.3a, we show a CGM in the form of Markov network and in the form of Bayesian network in Fig. 2.3b. Note that the two possible induced cycles in the graph go across 3 nodes, e.g.  $(x_1, x_2, x_3)$  and  $(x_4, x_5, x_6)$ . If an edge would exist between  $x_2$  and  $x_4$ , then a *chord* should also exist between  $x_5$  and  $x_2$  or between  $x_3$  and  $x_4$ , in order to keep the graph chordal.

An important property of chordal or decomposable models is that the graph of their maximal cliques is a tree, which is known as the *junction tree*, and hence the joint probability of a CGM can be expressed in the form,

$$p(\mathbf{x}; \boldsymbol{\theta}) = \frac{\prod_{c \in \mathcal{C}(T)} \psi_c(\mathbf{x}_c; \boldsymbol{\theta}_c)}{\prod_{s \in \mathcal{S}(T)} \psi_s(\mathbf{x}_s; \boldsymbol{\theta}_s)} \quad (2.12)$$

where  $\mathcal{C}(T)$  are nodes of the junction tree that contain the maximal cliques in the chordal graph and  $\mathcal{S}(T)$  are the *separators* of the tree. A *separator* is the set of variables that intersect between two neighbours in the junction tree. In Fig. 2.3,  $\mathcal{C}(T) = \{(x_1, x_2, x_3) (x_4, x_5, x_6) (x_3, x_5) (x_6, x_M)\}$  are the maximal cliques defining the junction tree nodes and  $\mathcal{S}(T) = \{(x_3) (x_5) (x_6)\}$  are the separators of the tree. Note that the joint probability for a CGM does not require to compute the normalising constant. Thus, the joint probability for the UGM in Fig. 2.3a can be written as,

$$p(\mathbf{x}; \boldsymbol{\theta}) = \frac{p(x_1, x_2, x_3; \theta_1^c) p(x_4, x_5, x_6; \theta_2^c) p(x_3, x_5; \theta_3^c) p(x_6, x_M; \theta_4^c)}{p(x_3; \theta_1^s) p(x_5; \theta_2^s) p(x_6; \theta_3^s)} \quad (2.13)$$

and its counterpart for the DGM in Fig. 2.3b can be derived through basic conditioning of probabilities. Another important property of chordal models to transform between representations is that a *perfect elimination ordering* can be found through LBFS (Lexicographic Breadth First Search). That is, a corresponding directed graph can be simply obtained with an algorithm that is linear to the cardinality of nodes and edges.

## 2.3 Text Representation

Text data is composed of words, numbers and other symbols separated by delimiters like white spaces or punctuation marks. In what follows, we use the term word to refer to any symbol within these delimiters. The arrangement of these words forms documents and their groupings, document collections or corpora. We consider documents exchangeable within the corpus, which means that any reordering of the corpus is equally likely to occur in probabilistic terms. The set of all unique words constitutes the vocabulary  $\mathcal{V} = \{v_1, \dots, v_V\}$ .

The  $n$ -gram representation considers contiguous sequences of  $n$  words from a text document or part of it. For instance, the sentence “What I cannot create” contains one 4-gram («What-I-cannot-create»), two Trigrams («What-I-cannot», «I-cannot-create»), three Bi-grams («What-I», «I-cannot», «cannot-create») and four Unigrams («What», «I», «cannot», «create»). Probability models build on  $n$ -grams of text, a.k.a. language models, suffer from the curse of dimensionality for large  $n$  because the probability space grows exponentially with the vocabulary size. Therefore, the simplifying Unigram model or bag of words is usually convenient for applications that do not require to exactly recover the original document, like in information retrieval or topic modelling.

### 2.3.1 Bag of Words Model

The Unigram model or bag of words, assumes that words are exchangeable within the document. This implies that the word order is lost and hence, a document is seen as a group or bag of words. Moreover, the bag of words model allows two possible representations depending on whether the selection order of words in the bag matters or not (Buntine and Jakulin, 2006).

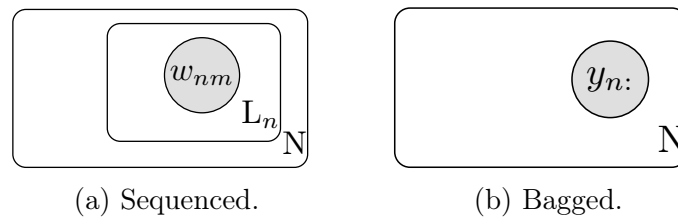


Figure 2.4: Bag-of-words or Unigram model.

In Fig. 2.4, we draw the two representations of the bag of words model in Plate notation. Due to the exchangeability of documents, both representations repeat the document random variables  $N$  times, i.e. the number of documents in the corpus. Representational differences arise within documents, where words can be:

- **Sequenced.** Words in the  $n$ -th document are represented as a sequence of  $L_n$  indexes in the vocabulary set  $\mathcal{V}$ , where  $L_n$  corresponds to the document length. Each word index is modelled as a Categorical random variable  $w_{nm}$  with  $V = |\mathcal{V}|$  possible outcomes, that is the vocabulary size. Fig. 2.4a shows that each r.v. in the  $n$ -th document is repeated  $L_n$  times. This representation disregards the possible word orders in the bag and hence, the probability of the  $n$ -th document can be expressed as,

$$p(\mathbf{w}_n; \mathbf{p}_n) = \prod_{m=1}^{L_n} \text{Cat}(w_{nm}; p_{nm}^s) \quad (2.14)$$

where  $\text{Cat}(\cdot)$  is a Categorical distribution as defined in Eq. (B.3) and  $p_{nm}^s$  is a probability vector over the vocabulary for the  $m$ -th word in the  $n$ -th document and  $\mathbf{p}_n^s = \{p_{n1}, \dots, p_{nL_n}\}$  is the set of probability vectors in this document.

- **Bagged.** Words in the  $n$ -th document are the total number of counts of that word in that document. Word counts are modelled as the number of successes of a Multinomial random variable  $\mathbf{y}_n$  with  $V$  categories and  $L_n$  trials. Fig. 2.4b shows that each document is a single Multinomial random variables  $\mathbf{w}_n$ . This representation accounts for all the possible word orders and hence, the probability of the  $n$ -th document can be expressed as,

$$p(\mathbf{y}_n; \mathbf{p}_n) = \text{Mult}(\mathbf{y}_n; L_n, \mathbf{p}_n) \quad (2.15)$$

where  $\text{Mult}(\cdot)$  is a Multinomial distribution as defined in Eq. (B.4) and  $\mathbf{p}_n^b$  is a probability vector over the vocabulary for all words in the  $n$ -th document.

Despite the differences, when the set of probability vectors in the sequenced representation  $\mathbf{p}_n^s$  are all equal to that in the bagged representation  $\mathbf{p}_n^b$ , the probability of the  $n$ -th document in the sequenced representation only differs from that in the bagged representation in a combinatoric term  $\frac{L_n!}{\prod_p y_{np}!}$ , which accounts for the word order in the latter.

To give an example, let us consider a corpus of a document whose content is: “What I cannot create I do not understand”. In this corpus, let us assume that the vocabulary is the set {What, I, cannot, create, do, not, understand, Richard, Feynman}, which include two more words not in the document. The sequenced representation of this document would be a list of indexes pointing at the words in the vocabulary [0, 1, 2, 3, 1, 4, 5, 6]. The bagged representation would be a vector of size 9, i.e. vocabulary size, with the corresponding word counts [1, 2, 1, 1, 1, 1, 1, 0, 0]. Then, every index in the sequenced representation is generated from an independent Categorical distribution over the vocabulary  $\mathcal{V}$ , whereas the count vector in the bagged representation is generated from a Multinomial distribution with  $V = 9$  categories and  $L_n = 8$  trials. Note that the sequenced representation imposes a specific word order and the bagged representation accounts for any possible order through the combinatoric term that in this example is  $8!/(1! 2! 1! 1! 1! 1! 1! 0! 0!) = 20160$ . This means that the same document in the bagged representation is 20160 times more likely than in the sequenced representation.

From the bagged representation above, one can also create other observational models by replacing the Multinomial distribution by other distributions. For instance, later we will introduce a model that uses independent Poisson distributions for each word to generate the word count vector. A more radical approach is to instead of modelling word counts,



simply capture the presence or absence of words. This is interesting because it exists an extensive literature on learning graphical models from binary data, despite the lost of information in binarisation. Therefore, one can develop a **binarised representation** for the presence/absence of words through  $V$  independent Bernoulli distributions as defined in Eq. (B.1). The result of this representation in the previous example is a binary vector of length  $L_n = 8$  such that  $[1, 1, 1, 1, 1, 1, 1, 0]$ .

## 2.4 Topic Models

Topic models (Blei, 2012) are algorithms for discovering the main themes of a document collection. Most topic models also rely on directed or undirected graphical models to described the statistical dependencies between their variables. Besides most of them represent text through a bag of words model given that the thematic information is preserved under this representation. Next, we review the most basic topic models that this thesis builds on and we classify them as per their type of PGM, their representation model and whether they fix or infer the number of topics and length. Fig. 2.1 shows a summary of this classification.

| Topic Model   | Type of PGM | Sequenced / Bagged | Count / Binary | Topics $K$ | Document Length $L_n$ |
|---|-------------|--------------------|----------------|------------|-----------------------|
| MoU (Nigam et al., 2000)                            | DGM/LVM     | Sequenced          | Count          | Fixed      | Fixed                 |
| MoB (Juan and Vidal, 2002)                          | DGM/LVM     | Bagged             | Binary         | Fixed      | Fixed                 |
| LDA (Blei et al., 2002)                             | DGM/LVM     | Sequenced          | Count          | Fixed      | Fixed                 |
| HDP (Teh et al., 2006b)                             | DGM/LVM     | Sequenced          | Count          | Inferred   | Fixed                 |
| mPCA (Buntine, 2002)                                | DGM/LVM     | Bagged             | Count          | Fixed      | Fixed                 |
| GaP (Canny, 2004)                                   | DGM/LVM     | Bagged             | Count          | Fixed      | Inferred              |
| $\beta\gamma\Gamma$ -PFA (Zhou et al., 2012)        | DGM/LVM     | Bagged             | Count          | Inferred   | Inferred              |
| BPFA (Zhou, 2015)                                   | DGM/LVM     | Bagged             | Binary         | Inferred   | Inferred              |
| HLTM (Chen et al., 2017)                            | DGM         | Bagged             | Binary         | Inferred   | Inferred              |
| RBM (Hinton, 2002)                                  | UGM         | Bagged             | Binary         | Fixed      | Inferred              |
| Replicated softmax (Hinton and Salakhutdinov, 2009) | UGM         | Sequenced          | Count          | Fixed      | Fixed                 |

Table 2.1: Summary of existing Topic Models for text.

### 2.4.1 Mixture of Unigrams (MoU)

The MoUs (Mixture of Unigramss) model (Nigam et al., 2000) assumes that the document corpus is composed of different subpopulations of documents. Each document belongs to ones of these subpopulations and documents in a subpopulation are generated from the same probability distribution. A document can be generated either from a sequence of  $L_n$  Categorical distributions, i.e. sequenced, or from a Multinomial distribution, i.e. bagged, as we show next.

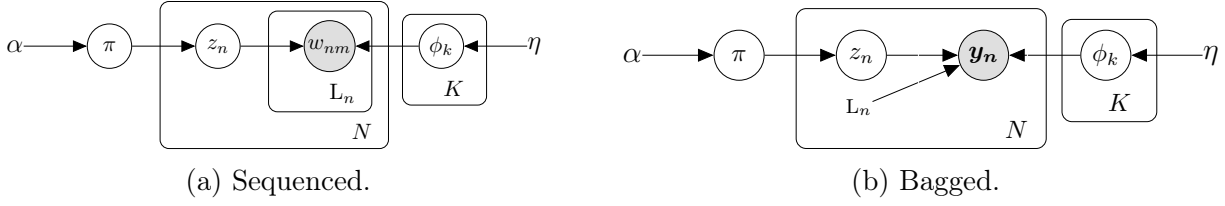


Figure 2.5: Mixture of Unigrams (MoU) graphical models.

Fig. 2.5 depicts the graphical model of MoU in the sequenced and bagged representations. MoU belongs to the subclass of LVM presented in Section 2.2.1.1. The document plate indexed by the number of documents  $N$  contains the local variables. In both representations, grey nodes,  $\{w_{n1}, \dots, w_{nL_n}\}$  and  $\mathbf{y}_n$ , refer to the observed words (sequenced) and counts (bagged) in the  $n$ -th document; and white node,  $z_n$ , to the latent variable that assigns the subpopulation to the  $n$ -th document. The global latent parameters for this model are in charge of modelling the proportions of subpopulations  $\pi$  and the  $K$  subpopulations  $\Phi = \{\phi_1, \dots, \phi_K\}$ . Each subpopulation  $\phi_k$  is a probability vector over the vocabulary  $\mathcal{V}$  which is usually referred as the  $k$ -th topic. Finally, the graphical model shows the hyperparameters as plain text linked to the variables or plates. We note that  $L_n$  is a distribution hyperparameter in the bagged representation whereas it is a model hyperparameter in the sequenced.

Following, the factorisation property of DGMs in Eq. (2.6), we can express the joint probability of MoUs in its bagged form as,

$$p(\mathbf{w}, \mathbf{z}, \pi, \Phi; \alpha, L_n, \eta) = p(\pi; \alpha) \prod_{n=1}^N p(z_n | \pi) p(\mathbf{y}_n | z_n, \Phi; L_n) \prod_{k=1}^K p(\phi_k; \eta) \quad (2.16)$$

where the observational model  $p(\mathbf{y}_n | z_n, \Phi; L_n)$  is  $\text{Mult}(\mathbf{y}_n; L_n, \phi_{z_n})$  and  $\phi_{z_n}$  indicates that variable  $z_n$  indexes the corresponding distribution in the set  $\Phi = \{\phi_1, \dots, \phi_K\}$ .

The sequenced version uses a different observational model. The conditional probability of the observed words given their parents is expressed through  $L_n$  independent distributions such that,

$$p(\mathbf{w}_n | z_n, \Phi; L_n) = \prod_{m=1}^{L_n} p(w_{nm} | z_n, \phi). \quad (2.17)$$

where each  $p(w_{nm} | z_n, \Phi)$  comes from a  $\text{Cat}(w_{nm}; \phi_{z_n})$ . The difference in probability between the sequenced and bagged representations is the same combinatoric term than in the bag of word model, given that  $\phi_{z_n}$  is shared across words. The graphical models above are complemented with the generative processes below that indicate the distributions used for each variable.

$\pi \sim \text{Dir}(\alpha)$   
 For each topic  $k = 1 \dots K$   
 $\phi_k \sim \text{Dir}(\eta)$   
 For each document  $n = 1 \dots N$   
 $z_n \sim \text{Cat}(\pi)$   
 For each word  $m = 1 \dots L_n$   
 $w_{nm} \sim \text{Cat}(\phi_{z_n})$

Process 2.1: Sequenced MoU.

$\pi \sim \text{Dir}(\alpha)$   
 For each topic  $k = 1 \dots K$   
 $\phi_k \sim \text{Dir}(\eta)$   
 For each document  $n = 1 \dots N$   
 $z_n \sim \text{Cat}(\pi)$   
 $\mathbf{y}_n \sim \text{Mult}(L_n, \phi_{z_n})$

Process 2.2: Bagged MoU.

In both representations, we observe that MoU considers Dirichlet priors, see definition in Eq. (B.8), for the topic proportions  $\pi$  and topic distributions  $\Phi$ . Besides the topic assignments  $z_n$  are also drawn from a Categorical distribution with proportions  $\pi$  in both. Thus, the differences occur in how they generate the observed words, from  $L_n$  Categorical distributions in the sequenced model or from a Multinomial in the bagged representation. As suggested earlier, one could also use an observational model with  $V$  Bernoullis to derive a **MoB (Mixture of Bernoullis)** (Juan and Vidal, 2002). MoU model is considered the most basic topic model since it only allows one topic per document.

### 2.4.2 Latent Dirichlet Allocation (LDA)

LDA (Latent Dirichlet Allocation) (Blei et al., 2002, 2003) extends the MoU topic model in the previous section to more than one topic per document. It does that by assuming that each word in a document is generated from a mixture of topics whose proportions are determined at the document level  $\Theta = \{\theta_1, \dots, \theta_N\}$ .

For each topic  $k = 1 \dots K$   
 $\phi_k \sim \text{Dir}(\eta)$   
 For each document  $n = 1 \dots N$   
 $\theta_n \sim \text{Dir}(\alpha)$   
 For each word  $m = 1 \dots L_n$   
 $z_{nm} \sim \text{Cat}(\theta_n)$   
 $w_{nm} \sim \text{Cat}(\phi_{z_{nm}})$

Process 2.3: LDA.

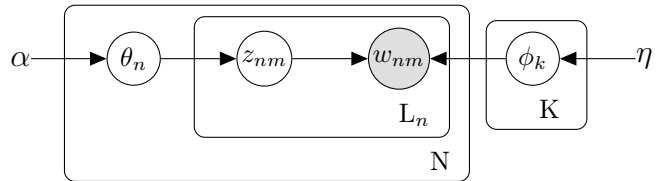


Figure 2.6: LDA graphical model.

That is to say, the  $n$ -th document can be summarised by its proportions over topics  $\theta_n$ , which correspond to the mixture proportions for generating each of the  $L_n$  observed words  $\{w_{n1}, \dots, w_{nL_n}\}$ . We note that LDA considers the sequenced representation of text introduced in Section 2.3.

From the graphical model in Fig. 2.6, we can write down the joint probability distribution

as follows,

$$p(\mathbf{w}, \mathbf{z}, \Theta, \Phi; \alpha, \eta) = \prod_{n=1}^N p(\theta_n; \alpha) \prod_{m=1}^{L_n} p(z_{nm}|\theta_n) p(w_{nm}|z_{nm}, \Phi) \prod_{k=1}^K p(\phi_k; \eta) \quad (2.18)$$

where the  $p(z_{nm}|\theta_n) = \text{Cat}(z_{nm}; \theta_n)$  and  $p(w_{nm}|z_{nm}, \Phi) = \text{Cat}(w_{nm}; \Phi_{z_{nm}})$  and Dirichlet priors on  $p(\theta_n; \alpha)$  and  $p(\phi_k; \eta)$ , as specified in Proc. 2.3.

The LDA model has been extended in many different directions. For instance, [Blei and Lafferty \(2006\)](#) introduced a dynamic topic model that uses a state-space model in the topics' plate to account for their evolution over time, i.e.  $\phi_{t,k}|\phi_{t-1,k} \sim \mathcal{N}(\phi_{t-1,k}, \sigma)$ . [Blei and Lafferty \(2007\)](#) also proposed to use a  $K$ -variate Normal prior over the proportions  $\Theta$  to capture the correlation between topics and overcome one of the main drawbacks of the Dirichlet prior. [Wallach \(2006\)](#) went beyond bag of words and proposed a bigram (i.e. 2-gram) language model to learn topics that are less dominated by function words (i.e. prepositions, conjunctions, etc.). Alternatively, [Wallach et al. \(2009a\)](#) studied the impact that has the use of symmetric and asymmetric Dirichlet priors for topics and their proportions. While they found that the use of asymmetric priors over the topic proportions, i.e.  $\alpha = (\alpha_1, \dots, \alpha_K)$ , improved the model performance, they reported that symmetric Dirichlet priors were sufficient for the topic distributions, i.e.  $\eta = (\eta, \dots, \eta)$ .

However, MoU, LDA and many of these extensions still require to specify the number of topics  $K$  beforehand, limiting the exploration capabilities of these algorithms in unknown document collections.

### 2.4.3 Hierarchical Dirichlet Process (HDP)

The number of topics  $K$  in LDA has to be fixed in advance and this might compromise the performance of the topic model. Although cross-validation can be used to determine the right number of topics, nonparametric models such as the HDP (Hierarchical Dirichlet Process) ([Teh et al., 2006b](#)) have been introduced to address this limitation. These nonparametric models can be framed in the context of LVMs with stochastic processes as priors, e.g. a hierarchical DP (Dirichlet Process) is used as prior in a LDA-like topic model.

The DP ([Ferguson, 1973](#)) is a distribution over probability measures  $G : \Theta \rightarrow \mathbb{R}^+$  such that  $G(\theta) \geq 0$  and  $\int_{\Theta} G(\theta) d\theta = 1$ . That is to say that any realization of a DP is a probability distribution over the space  $\Theta$ . Moreover, any finite partition of the space  $\Theta = \bigcup_{k=1}^K T_k$  is jointly distributed according to a Dirichlet distribution parametrised as,

$$G(T_1), \dots, G(T_K) \sim \text{Dir}(\alpha H(T_1), \dots, \alpha H(T_K)) \quad (2.19)$$

where  $\alpha$  is a positive real number called the concentration parameter and  $H$ , the base measure. A Dirichlet process with this parametrisation is often expressed as  $\text{DP}(\alpha, H)$ . [Sethuraman \(1994\)](#) provided a constructive definition of DPs in which a draw from  $G \sim \text{DP}(\alpha, H)$  is, with probability one,

$$G_0 = \sum_{k=1}^{\infty} \pi_k \delta_{\theta_k} \quad (2.20)$$

where  $\theta_k \sim H$ ,  $\delta_{\theta_k}$  represents an atom at  $\theta_k \in \Theta$  and  $\pi_k \sim \text{GEM}(\alpha)$  are the proportions drawn from the distribution underlying the stick-breaking process, see Appendix B.4. DPs

can be used as priors for mixture models to infer the number of mixture components. For example, one can simply derive a non-parametric extension of the MoU model in Section 2.4.1 by using a  $DP(\alpha, H)$  where  $H$  is a Dirichlet distribution over the vocabulary and  $\alpha$  the same Dirichlet parameter. Then, the mixture proportions are generated from the  $GEM(\alpha)$  and words, from the corresponding atoms, i.e. topics.

$G_0 \sim DP(\gamma, H)$   
 For each document  $n = 1 \dots N$   
 $G_n \sim DP(\alpha_0, G_0)$   
 For each word  $m = 1 \dots L_n$   
 $\theta_{nm} \sim G_n$   
 $w_{nm} \sim \text{Cat}(\theta_{nm})$

Process 2.4: HDP.

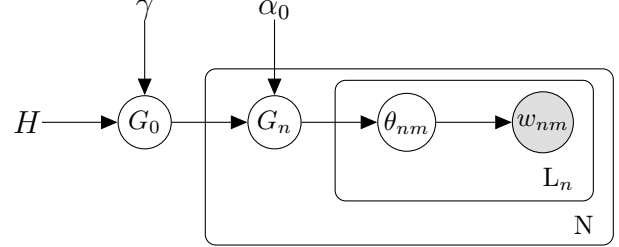


Figure 2.7: HDP graphical model.

The HDP is a hierarchical construction of Dirichlet processes to enable multiple topics in the same document as in LDA. The top-layer of the model is a process of the form  $G_0 \sim DP(\gamma, H)$ , where  $\gamma$  is the concentration parameter that controls the amount of variability around  $H$ , which is the base measure on the probability vectors that encode the topics. The next layer is then constructed using the realization of the process  $G_0$  as base measure for the document-level DP and  $\alpha_0$  as concentration,  $G_n \sim DP(\alpha_0, G_0)$ . Therefore,  $\alpha_0$  controls how much the  $G_n$  process deviates from  $G_0$ . This hierarchy enables topics, which are the atoms of the top-layer DP, to be shared across documents, but each exhibits different proportions according to the proportions drawn from  $G_n$ . Finally, words in the  $n$ -th document are generated in a sequenced fashion from the corresponding topic distributions. The generative process is detailed in Proc. 2.4 and the graphical representation in Fig. 2.7. Although it is omitted in the above descriptions, the HDP model, like LDA, often considers a Dirichlet prior over the topics such that  $H \sim \text{Dir}(\eta)$ .

In conclusion, the result of using a hierarchy of Dirichlet processes enables HDP to learn a posterior distribution over the number of topics  $K$ , as it is done for any other latent parameter in a probability model. Through this posterior, one can set the number of topics to its mean, mode or any other value related to this distribution.

#### 2.4.4 Multinomial Principal Component Analysis (mPCA)

mPCA (Multinomial Principal Component Analysis) (Buntine, 2002; Buntine and Jakulin, 2004) was presented around the same time as LDA as a discrete extension of PCA (Principal Component Analysis).

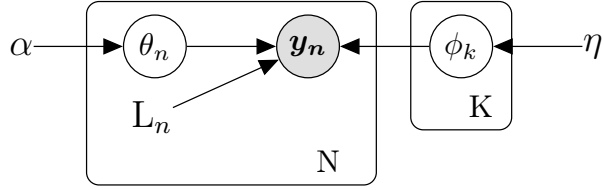
For each topic  $k = 1 \dots K$

$$\phi_k \sim \text{Dir}(\eta)$$

For each document  $n = 1 \dots N$

$$\theta_n \sim \text{Dir}(\alpha)$$

$$\mathbf{y}_n \sim \text{Mult}(\theta_n \Phi)$$



Process 2.5: mPCA.

Figure 2.8: mPCA graphical model.

In contrast to LDA, the model is built on the bagged representation of text by assuming that words in the  $n$ -th document are generated from  $\text{Mult}(\mathbf{y}_n; L_n, \theta_n \Phi)$ , where the event probabilities come from the product of two matrices  $\Theta$  ( $N \times K$ ) and  $\Phi$  ( $K \times V$ ). Similar to LDA, the model considers the same prior distributions over the proportion  $\theta_n$  and over the topic distributions  $\phi_k$ . From Fig. 2.8, the joint probability distribution can be written as,

$$p(\mathbf{y}, \Theta, \Phi; \alpha, \eta) = \prod_{n=1}^N p(\theta_n; \alpha) p(\mathbf{y}_n | \theta_n, \Phi; L_n) \prod_{k=1}^K p(\phi_k; \eta) \quad (2.21)$$

where the term  $p(\mathbf{y}_n | \theta_n, \Phi; L_n)$  is given by the Multinomial distribution above. Therefore, the main difference between LDA and mPCA is in the observational model, where the sequenced representation is considered in LDA and the bagged representation, in mPCA. We can see that if we collapse the topic assignments  $\mathbf{z}$  in the LDA model, their joint distributions are related by the combinatorial term that accounts for the multiple ordering of words in a document,

$$p^{\text{mPCA}}(\mathbf{y}, \Theta, \Phi; \alpha, \eta) = \prod_{n=1}^N \frac{L_n!}{\prod_m y_{np}!} p^{\text{LDA}}(\mathbf{w}, \Theta, \Phi; \alpha, \eta). \quad (2.22)$$

where  $y_{np}$  on the right hand side indicates the total counts of the  $p$ -th word in the vocabulary in the  $n$ -th document.

Given that the combinatoric term does not depend on the parameters, the evidence of both models only differ on this combinatoric term. Thus, the relationship in terms of the posterior distribution cancels out this term and hence, both models has the same posterior.

$$p^{\text{mPCA}}(\Theta, \Phi | \mathbf{y}; \alpha, \eta) = p^{\text{LDA}}(\Theta, \Phi | \mathbf{w}; \alpha, \eta) \quad (2.23)$$

This mean that the representation of text in the sequenced or bagged forms is irrelevant when the fitting of LDA and mPCA (Buntine and Jakulin, 2006). However, this bagged representation in mPCA is interesting to introduce a fully generative class of models that do not condition on the document length.

### 2.4.5 Gamma Poisson (GaP)

GaP (Gamma Poisson) was proposed by Canny (2004) to address the fact that previous models did not represent the uncertainty over the document length. To achieve that, GaP builds on the mPCA factorisation but it instead considers a Poisson observational model.

The observational model  $p(\mathbf{y}_n | \mathbf{l}_n, \Phi)$  is the product of  $V$  Poisson distributions, each with rate  $\lambda_{nm} = \sum_k l_{nk} \phi_{km}$ . Furthermore, a Gamma prior is considered for each parameter  $l_{nk} \sim \text{Ga}(\alpha_k, \beta_k)$ , as defined in (B.9). In contrast to LDA and mPCA, the original GaP did not consider a prior over  $\Phi$ , but subsequent extensions did use a Dirichlet prior too, as we show in Proc. 2.6.

For each topic  $k = 1 \dots K$

$$\phi_k \sim \text{Dir}(\eta)$$

For each document  $n = 1 \dots N$

For each topic  $k = 1 \dots K$

$$l_{nk} \sim \text{Ga}(\alpha_k, \beta_k)$$

For each word  $p = 1 \dots V$

$$y_{np} \sim \text{Pois}\left(\sum_k l_{nk} \phi_{kp}\right)$$

Process 2.6: GaP (I).

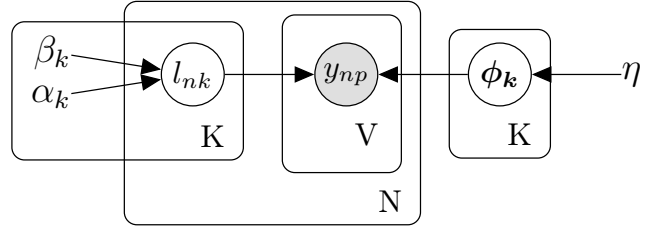


Figure 2.9: GaP graphical model (I).

The joint probability for GaP can also be written down from the graphical model in Fig. 2.9 as,

$$p(\mathbf{y}, L, \Phi; \alpha, \beta, \eta) = \prod_{n=1}^N \prod_{k=1}^K p(l_{nk}; \alpha, \beta) \prod_{p=1}^V p(y_{np} | \mathbf{l}_n, \Phi_{:p}) \prod_{k=1}^K p(\phi_k; \eta) \quad (2.24)$$

where the main difference to mPCA is not only the Gamma priors over the  $K$ -th components of  $\mathbf{l}_n$ , but also the Poisson likelihoods over the  $V$  words in the vocabulary without any conditioning to the sum of their counts, i.e. the document length.

For each topic  $k = 1 \dots K$

$$\phi_k \sim \text{Dir}(\eta)$$

For each document  $n = 1 \dots N$

For each topic  $k = 1 \dots K$

$$l_{nk} \sim \text{Ga}(\alpha_k, \beta_k)$$

$$L_n \sim \text{Pois}\left(\sum_k l_{nk}\right)$$

$$\mathbf{y}_n \sim \text{Mult}\left(L_n, \sum_k \frac{l_{nk}}{\sum_{k'} l_{nk'}} \phi_{k:}\right)$$

Process 2.7: GaP (II).

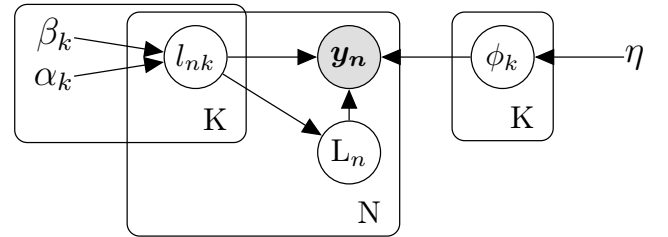


Figure 2.10: GaP graphical model (II).

An equivalent representation to GaP can be derived from a property which relates the Poisson and Multinomial distributions. Given that  $\{w_{nm} \forall m = 1 \dots V\}$  are independent Poisson random variables with rates  $\sum_k l_{nk} \phi_{km}$ , the joint probability of these variables can

be expressed as,

$$p(\mathbf{y}_n | \mathbf{l}_n, \Phi) = \text{Mult}(\mathbf{y}_n; L_n, \sum_k \frac{l_{nk}}{\sum_{k'} l_{nk'}} \phi_k) \text{Pois}(L_n; \sum_k l_{nk}) \quad (2.25)$$

where  $L_n$  is the latent document length which corresponds to the sum of  $V$  independent Poisson random variables. Fig. 2.10 and Proc. 2.7 depicts the GaP graphical model under this new representation which is more explicit on how the uncertainty over the document length is modelled.

Two properties of the Gamma distribution allowed [Buntine and Jakulin \(2006\)](#) to draw connections between GaP, mPCA and LDA. When  $K$  independent random variables are drawn from  $l_{nk} \sim \text{Ga}(\boldsymbol{\alpha}, \beta)$  and their values are normalised by their sum  $\frac{l_n}{\sum_{k'} l_{nk'}}$ , the resulting random variable follows a Dirichlet distribution with parameter  $\boldsymbol{\alpha} = \{\alpha_1, \dots, \alpha_K\}$ . Besides the sum of these  $K$  random variables,  $\sum_k l_{nk}$ , is distributed according to a Gamma distribution with shape  $\sum_k \alpha_k$  and scale  $\beta$ . This implies that  $\theta_n = \frac{l_n}{\sum_{k'} l_{nk'}}$  in Eq. (2.25) is now distributed according to  $\text{Dir}(\boldsymbol{\alpha})$  when the hyperparameters  $\beta_k$  are kept constant to  $\beta$  for all  $k$ . Furthermore, the resulting observational model, after integrating out  $\sum_k l_{nk}$ , can be written as,

$$p(\mathbf{y}_n | \theta_n, \Phi) = \text{Mult}(\mathbf{y}_n; L_n, \theta_n \Phi) \text{NB}(L_n; \sum_k \alpha_k, \frac{\beta}{1 + \beta}) \quad (2.26)$$

where  $\theta_n \sim \text{Dir}(\boldsymbol{\alpha})$  and the marginalized Poisson-Gamma composition produces a NB (Negative Binomial) distribution as in Eq. (B.12). Given that the NB distribution on the document length only depends on the hyperparameters, the GaP model becomes equivalent to mPCA and LDA ignoring representational issues.

Finally, a property of Poisson random variables allows the expression of GaP in a way that the assignment of words to topics is made explicit through a latent count variable. This representation will also be useful for introducing the non-parametric extensions. A Poisson random variable with rate  $\lambda_{nm} = \sum_k l_{nk} \phi_{km}$  is equivalent to the sum of  $K$  Poisson random variables, each with rate  $\lambda_{nmk} = l_{nk} \phi_{km}$ . As a results, we can augment the model in Fig. 2.9 with latent variables  $x_{nmk} \sim \text{Pois}(\lambda_{nmk})$ , and express the observed counts as the sum of these latent variables  $y_{nm} = \sum_k x_{nmk}$ . These latent variables represent the number of times that topic  $k$  has been assigned to  $m$ -th word in the  $n$ -th document.



For each topic  $k = 1 \dots K$   
 $\phi_k \sim \text{Dir}(\eta)$   
 For each document  $n = 1 \dots N$   
 For each topic  $k = 1 \dots K$   
 $l_{nk} \sim \text{Ga}(\alpha_k, \beta_k)$   
 For each word  $p = 1 \dots V$   
 For each topic  $k = 1 \dots K$   
 $x_{npk} \sim \text{Pois}(l_{nk}\phi_{kp})$   
 $y_{np} = \sum_k x_{npk}$

Process 2.8: GaP (III).

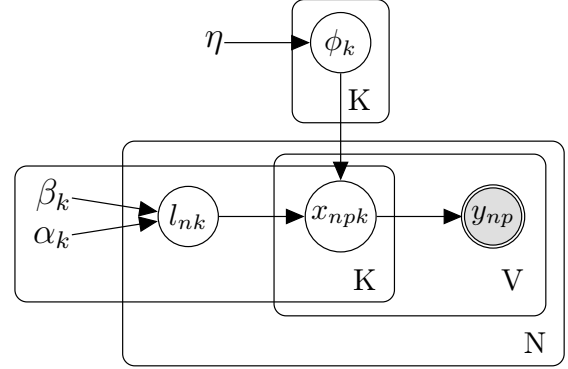


Figure 2.11: GaP graphical model (III).

Proc. 2.8 and Fig. 2.11 show the generative process and graphical model for this augmented model, respectively. We note that the relationship between observed counts  $y_{np}$  and latent counts  $x_{npk}$  is deterministic, as displayed by the double-lined node.

GaP can also be extended with non-parametric prior that enable to infer the number of topics. However, the fact that in the general case its latent variables  $\mathbf{l}_n$  are not normalised by their sum, requires to use a different types of stochastic processes to achieve similar capabilities to the HDP.

### 2.4.6 Beta-Negative Binomial Process (BNB)

The number of topics in GaP is a model hyperparameter that has to be fixed beforehand. To address this in a non-parametric fashion, we need to introduce the BNB (Beta-Negative Binomial Process) (Zhou et al., 2012), which is constructed from coupling a NB with a marked BP (Beta Process).

The BP (Hjort et al., 1990; Thibaux and Jordan, 2007) is a positive Lévy process whose Lévy measure is defined on the product space  $[0, 1] \times \Omega$  as,

$$\nu_{BP}(dpd\omega) = cp^{-1}(1-p)^{c-1}dpB_0(d\omega) \quad (2.27)$$

where  $c > 0$  is called the concentration parameter,  $B_0$  is a continuous measure over  $\Omega$  called the base measure, and  $\alpha = B_0(\Omega)$  is the mass parameter. Then, the points from this process,  $(w_k, p_k) \in [0, 1] \times \Omega$ , can be marked with a random variable  $r_k$  taking values in  $\mathbb{R}^+$  where  $r_k$  and  $r'_k$  are independent for  $k \neq k'$ . This leads to a marked BP with Lévy measure defined on the product space  $[0, 1] \times \mathbb{R}^+ \times \Omega$  as,

$$\nu_{BP}^*(dpdrd\omega) = cp^{-1}(1-p)^{c-1}dpR_0(dr)B_0(d\omega) \quad (2.28)$$

where  $R_0$  is a continuous finite measure over  $\mathbb{R}^+$  with mass parameter given by  $\gamma = R_0(\mathbb{R}^+)$ . To sample from this marked process,  $B^* \sim BP(c, R_0B_0)$  one can draw a set of points  $(p_k, r_k, \omega_k)$  from a Poisson process with mean measure  $\nu_{BP}^*$  and express,

$$B^* = \sum_{k=1}^{\infty} p_k \delta_{r_k, \omega_k} \quad (2.29)$$

where  $\delta_{r_k, \omega_k}$  is the atom at  $(r_k, \omega_k) \in [0, 1] \times \mathbb{R}^+$  and  $p_k \in [0, 1]$  is the corresponding weight. Note that these weights do not have to be normalised as in the DP in Section 2.4.3.

Finally, a draw from a NBP (Negative Binomial Process),  $\text{NBP}(B^*)$  is defined as,

$$X_i = \sum_{k=1}^{\infty} \kappa_{ik} \delta_{\omega_k}, \quad \kappa_{ik} \sim \text{NB}(r_k, p_k) \quad (2.30)$$

where the  $i$ -th count  $\kappa_{ik}$  drawn from the  $\text{NB}(r_k, p_k)$  is associated to the atom  $\delta_{\omega_k}$ . This construction is similar to that of the Bernoulli Process (BeP) (Thibaux and Jordan, 2007), but  $\kappa_{ik}$  are related to counts rather than binary values.

However, the above process is not finite and it leads to countably infinite points. Thus, a finite approximation to the marked Beta process above is usually considered in practice. The modified Lévy measure for the finite approximation is given by,

$$\nu_{\epsilon BP}^*(dpdrd\omega) = cp^{\epsilon-1}(1-p)^{c(1-\epsilon)-1}dpR_0(dr)B_0(d\omega) \quad (2.31)$$

where  $\epsilon > 0$  is introduced to ensure that the measure on  $[0, 1] \times \mathbb{R}^+ \times \Omega$  is finite with value,

$$\nu_{\epsilon BP}^+ = \nu_{\epsilon BP}^*([0, 1] \times \mathbb{R}^+ \times \Omega) = c\gamma\alpha B(c\epsilon, c(1-\epsilon)). \quad (2.32)$$

Zhou et al. (2012) incorporated this finite approximation of the BP as the base measure for a NBP to construct an  $\epsilon$ BNB. This finite process was then used as the non-parametric prior for the latent counts in the extended GaP model described in Proc. 2.8. Furthermore, they placed a Gamma prior over the  $r_k$  to build a Beta-Gamma-Gamma-Poisson model named  $\beta\gamma\Gamma$ -PFA, where the hierarchy Beta-Gamma-Gamma acts as a prior of PFA (Poisson Factor Analysis). Proc. 2.9 and Fig. 2.12 show its generative process and graphical model.

$$K \sim \text{Pois}(\nu_{\epsilon BP}^+)$$

For each topic  $k = 1 \dots K$

$$\phi_k \sim \text{Dir}(\eta)$$

$$p_k \sim \text{Beta}(c\epsilon, c(1-\epsilon))$$

$$r_k \sim \text{Ga}(c_0 r_0, 1/c_0)$$

For each document  $n = 1 \dots N$

For each topic  $k = 1 \dots K$

$$l_{nk} \sim \text{Ga}(r_k, \frac{p_k}{1-p_k})$$

For each word  $p = 1 \dots V$

For each topic  $k = 1 \dots K$

$$x_{npk} \sim \text{Pois}(l_{nk}\phi_{kp})$$

$$y_{np} = \sum_k x_{npk}$$

Process 2.9:  $\beta\gamma\Gamma$ -PFA.

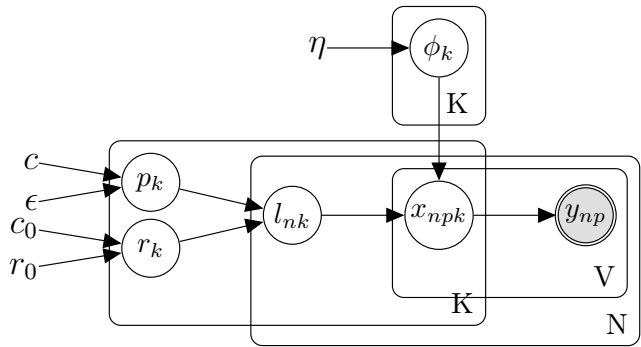


Figure 2.12:  $\beta\gamma\Gamma$ -PFA graphical model.

A wide range of topic models for binarised text can be constructed from combining the BerPo (Bernoulli-Poisson) Link presented in (Zhou, 2015) with PFA models in (Zhou

et al., 2012), which it is called BPFA (Bernoulli PFA). The BerPo link is a function which threshold the the counts  $y_{np}$  to presence (“1”) or absence (“0”). Mathematically, we can express this function as,

$$b_{np} = \mathbf{1}(y_{np}), \quad y_{np} \sim \text{Pois}(\lambda_{np}) \quad (2.33)$$

where  $b_{np} = 1$  if  $y_{np} \geq 1$ , and  $b_{np} = 0$ , otherwise; and  $\lambda_{np}$  refers to the corresponding Poisson rate, which is  $l_{nk}\phi_{kp}$  for the model above. However, the counts  $y_{np}$  are not observed in the BPFA model, because it assumes a binarised representation of text. Therefore, one might be interested in deriving the marginal distribution that integrates out the latent counts  $y_{np}$ . It turns out that this distribution is,

$$b_{np} \sim \text{Ber}(1 - e^{-\lambda_{np}}) \quad (2.34)$$

a Bernoulli distribution with probability parameter as defined in Eq. (B.1). Another important property to derive inference algorithms for this model is that the conditional probability of the latent counts given the presence/absence of words can be expressed as,

$$y_{np}|b_{np}, \lambda_{np} \sim b_{np} \text{Pois}_+(\lambda_{np}) \quad (2.35)$$

where  $\text{Pois}_+(\cdot)$  is a zero-truncated Poisson distribution over the positive integers as defined in Eq. (B.6) and  $b_{np}$  is in charge of forcing the zeros.

### 2.4.7 Latent Tree Models (LTMs)

LTMs (Latent Tree Models) (Choi et al., 2011) are rooted tree-structured graphical models with leaf nodes corresponding to the observed variables and internal nodes, to the latent or hidden variables, see an example of graphical model in Fig. 2.13. In contrast to the previous models they do not necessarily have a recurrence that enables their expression as template models. They represent a more general class of probability distributions than fully-observed trees due to the presence of latent variables, but they preserve some of the computational advantages, like efficient exact inference.

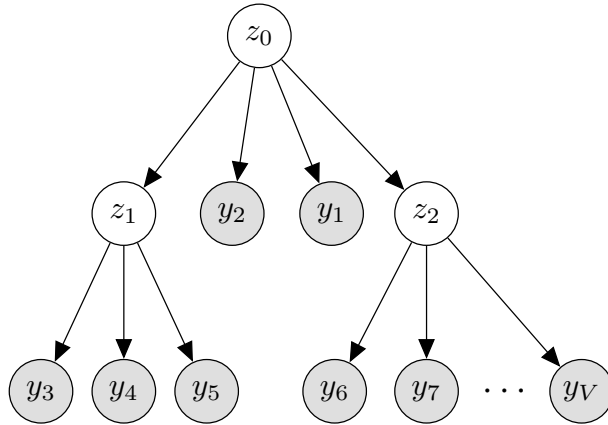


Figure 2.13: Latent Tree Models (LTMs).

The joint probability distribution of LTMs can be compactly represented as,

$$p(\mathbf{y}, \mathbf{z}) = p(x_r) \prod_{u \rightarrow v} p(x_v | x_u); \quad \forall x \in \mathcal{Y} \cup \mathcal{Z} \quad (2.36)$$

where  $\mathcal{Y}$  corresponds to the set of observed variables, and  $\mathcal{Z}$  to the set of latent variables. In contrast to the general probability rule for directed graphical models in Eq. (2.6), tree-structured graphs only have one parent for each variable in the model.

Lately, latent trees have been used to build hierarchies of topics in text (Chen et al., 2017), the so-called HLTMs (Hierarchical Latent Tree Models). In HLTMs, the observed variables represent the presence or absence of words while the hidden variables represent the unknown topics. Therefore, they use a binary and bagged representation of text. A topic is represented through the subset of observed words that the latent variable connects to, whereas in previous models, topics were distributions over the whole vocabulary. As a result, latent variables at high levels of the tree provide more general topics than those at lower levels, because they connect to more latent variables and they are able to capture long-range dependencies.

One of the main limitations of HLTMs is that text data needs to be binarised in order to learn the structure of latent trees, losing the representational power of count data. This is due to the fact that most structure learning algorithms only work for categorical data, not for count data.

### 2.4.8 Restricted Boltzmann Machines (RBMs)

RBMs (Restricted Boltzmann Machines) (Hinton, 2002) are undirected models with a bipartite graph of latent and observed variables. Observed variables are usually placed at the bottom and the layer of latent variables stack on top. No edge is allowed among observed nor among latent, but layers are usually fully-connected. Therefore, the joint distribution can be written as,

$$p(\mathbf{y}, \mathbf{z}; \boldsymbol{\theta}) = \frac{1}{Z(\boldsymbol{\theta})} \prod_{m=1}^V \prod_{k=1}^K \psi_{mk}(y_m, z_k | \boldsymbol{\theta}_{mk}) \quad (2.37)$$

following the Hammersley-Clifford theorem and the graphical model in Fig. 2.14, where  $\mathbf{y}$  is the set of observed variables,  $\mathbf{z}$  is the set of latent variables and  $\psi_{mk}$  the potential functional between every pair of observed and latent nodes. One can show that the partition function,  $Z(\boldsymbol{\theta})$ , becomes quickly intractable even for a few binary hidden variables.

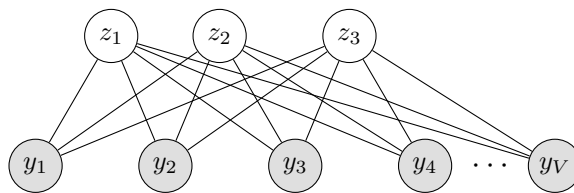


Figure 2.14: Restricted Boltzmann Machines (RBMs).

The main difference between RBMs and directed latent variable models like GaP is that latent variables in RBM are mutually independent given the observed words. This makes inference easier than in directed latent models, since each latent variable  $z_k$  can be estimated independently. Moreover, the fact that RBMs multiply a set of potential functions, instead of computing a mixture of these potentials, enables sharper distributions. For example, undirected topic models with RBMs (Hinton and Salakhutdinov, 2009) are capable of giving

high probability to words like “Berlusconi”, which are not high probable word in any of the topics present in the document (e.g. *government*, *mafia* and *playboy*), whereas mixture-based topic models like LDA or GaP smooth the probability of “Berlusconi” because of the averaging effect of the mixture.

## 2.5 Bayesian Inference

In the previous section, we have presented several probability models that encode the assumptions about the data generating process. To reverse the process, statistical inference uncovers properties of the underlying distribution by going from data to distributions. In particular, we focus here on Bayesian inference, which is a particular case of inference that makes use of the Bayes’ theorem to reverse the process. The Bayes’ rule applied to statistical inference can be written as,

$$p(\theta|\mathbf{x}) = \frac{p(\mathbf{x}|\theta)p(\theta)}{p(\mathbf{x})} \quad (2.38)$$

where  $p(\theta)$  is the prior distribution over the parameters of the model  $\theta$ ,  $p(\mathbf{x}|\theta)$  is the likelihood of the data  $\mathbf{x}$  under the model parametrised by  $\theta$  and  $p(\mathbf{x})$  is the model evidence or marginal likelihood. Bayes’ rule relates these three probabilities with the posterior probability  $p(\theta|\mathbf{x})$ , which represents the uncertainty over the parameters after having observed the data  $\mathbf{x}$ . In contrast, MAP (Maximum a Posteriori) inference outputs the mode of the posterior distribution.

Computing the posterior distribution plays a central role in Bayesian inference. First, it enables to uncover hidden structure in the data (i.e. clusters, topics, etc.) through the posterior distribution over certain model parameters. For example, the posterior distribution  $p(\Phi|\mathbf{w})$  in the LDA or mPCA model from Section 2.4.2 contain the underlying topic distributions in this corpus. Second, the probability of unseen data given the probability model can be computed by averaging over the posterior distribution as,

$$p(\mathbf{x}^*|\mathbf{x}) = \int p(\mathbf{x}^*|\theta, \mathbf{x})p(\theta|\mathbf{x}) \mathrm{d}\theta \quad (2.39)$$

where  $p(\mathbf{x}^*|\theta, \mathbf{x})$  is the likelihood of an unseen datum  $\mathbf{x}^*$  and  $p(\mathbf{x}^*|\mathbf{x})$  is its posterior predictive distribution. This quantity enables to quantify the uncertainty over a prediction as well as to compare the performance of different models in terms of their predictability.

However, computing a closed-form and tractable expression for the posterior distribution is only feasible for simple models with conjugacy like the Bernoulli-Beta or Poisson-Gamma models discussed earlier. In most interesting models, conjugacy does not hold and posterior computation becomes intractable due to the normalising factor in the denominator of Eq. (2.38), which consist of,

$$p(\mathbf{x}) = \int p(\mathbf{x}|\theta)p(\theta) \mathrm{d}\theta \quad (2.40)$$

an integration (or sum) over all model variables. In the next sections, we will present two basics approaches used to address this problem in approximate manner, which will be used throughout the thesis.

### 2.5.1 Gibbs Sampling

Gibbs sampling (Geman and Geman, 1984) is a MCMC (Markov Chain Monte Carlo) method to sample from high-dimensional distributions, where simpler Monte Carlo methods like rejection sampling have difficulties to draw samples. To address this, MCMC methods construct a Markov Chain on the state space whose equilibrium or stationary distribution is the desired one. That is to say that the fraction of visits in a given state is proportional to the target density.

Gibbs sampling performs this through an iterative scheme in which each variable is sampled sequentially according to its complete conditional,  $p(x'_i | \mathbf{x}_{\setminus i}^{(s)})$ . The complete conditionals are built by conditioning on the previous samples and the process is repeated until reaching equilibrium. After that, samples, supposedly coming from the desired distribution, are recorded during several more cycles to build an empirical posterior distribution.

Gibbs sampling is a particular case of a more general algorithm called MH (Metropolis Hastings). The MH algorithm is also an iterative algorithm in which one decides at each step to move from one state  $\mathbf{x}$  to another  $\mathbf{x}'$  with probability  $q(\mathbf{x}' | \mathbf{x})$ . In Gibbs sampling, the proposal distribution  $q(\mathbf{x}' | \mathbf{x})$  that modulates the jump from state  $\mathbf{x}$  to state  $\mathbf{x}'$  has the following form,

$$q(\mathbf{x}' | \mathbf{x}) = p(x'_i | \mathbf{x}_{\setminus i}) \mathbb{I}(\mathbf{x}'_{\setminus i} = \mathbf{x}_{\setminus i}) \quad (2.41)$$

where  $p(x'_i | \mathbf{x}_{\setminus i})$  is the complete conditional distribution for variable  $i$  and the term  $\mathbb{I}(\mathbf{x}'_{\setminus i} = \mathbf{x}_{\setminus i})$  indicates that all other variables except  $x_i$  are left unchanged.

Despite this proposal distribution has 100% acceptance rate, the fact that each variable is sampled sequentially does not imply that this method converges faster than others. However, this simple scheme makes Gibbs sampling very popular and general to use it for different types of models. This is the reason of its use in software packages such as BUGS (Lunn et al., 2012) and JAGS (Hornik et al., 2003), that automatize the inference for a great variety of probabilistic models.

Moreover, Gibbs sampling leverages on the structure of graphical models to reduce the number of variables that the complete conditionals depend on. These are the variables in the Markov Blanket of  $x_v$  in a graph  $\mathcal{G}$ , which for undirected graphs are its immediate neighbours and for directed, the union set of its children, parents and co-parents. For instance, the  $z_n$  variable in the LVM in Fig. 2.2 is conditionally independent of  $\mathbf{z}_{\setminus n}$  given its parent  $\phi$  and child  $x_n$ . Thus, one can express the complete conditional for the  $n$ -th local latent variable, which is assumed to be in the exponential form for the conditionally conjugate LVM, as,

$$p(z_n | x_n, \phi) = h(z_n) \exp(\eta_l(x_n, \phi) t(z_n) - a(\eta_l(x_n, \phi))) \quad (2.42)$$

where  $\eta_l(x_n, \phi)$  is the corresponding vector of natural parameters corresponding to  $z_n$  and the expression also reveals the independence between  $z_n$  and  $\mathbf{z}_{\setminus n}$  when  $\phi$  is observed. Similarly, the complete conditional for the global latent parameters  $\phi$  can be written as,

$$p(\phi | \mathbf{x}, \mathbf{z}; \gamma) = h(\phi) \exp(\eta_g(\mathbf{x}, \mathbf{z})^T t(\phi) - a(\eta_g(\mathbf{x}, \mathbf{z}))) \quad (2.43)$$

where  $\eta_g(\mathbf{x}, \mathbf{z})$  is the vector of global natural parameters corresponding to  $\phi$  and the dependence among all the local contexts is also highlighted by this expression.

Therefore, one can sample the posterior distribution of the LVM in Fig. 2.2 through a Gibbs sampling algorithm that iteratively samples the global complete conditional in

Eq. (2.43) and then the  $N$  complete conditionals for the local variables in Eq. (2.42). As mentioned earlier, the initial samples are discarded until the Markov Chain has converged to equilibrium. After this burn-in period, samples are collected to compute statistics of the posterior distribution. However, determining the duration of the burn-in period as well as when to stop the sampling are known drawbacks of MCMC methods. In the following section, we will introduce a deterministic method that does not suffer from this drawback.

### 2.5.2 Mean-field Variational Inference

Mean-field variational inference (Wainwright et al., 2008) approximates the posterior distribution by a family of distributions over the latent variables which is optimised to be close to the posterior.

The mean-field variational family assumes that latent variables in the model are independent and each variable is controlled by its own variational parameters. In the LVM from Fig. 2.2, the mean-field variational family can be written as,

$$q(\phi, \mathbf{z}|\Omega) = q(\phi|\lambda) \prod_n q(z_n|\psi_n) \quad (2.44)$$

where  $\Omega = \{\lambda, \boldsymbol{\psi}\}$  are the set of variational parameters and  $q()$  are the mean-field distributions governing each latent variable.

The optimisation framework to approximate  $p(\phi, \mathbf{z}|\mathbf{w}; \gamma)$  with the variational family  $q(\phi, \mathbf{z}|\Omega)$  is formulated in terms of minimizing the KL (Kullback-Leibler) divergence between both distributions,

$$\Omega^* = \arg \min_{\Omega} \text{KL}(q(\phi, \mathbf{z}|\Omega), p(\phi, \mathbf{z}|\mathbf{w}; \gamma)). \quad (2.45)$$

However, this objective cannot be directly optimised because it depends on the unknown posterior, but one can maximise a surrogate known as the ELBO (Evidence Lower BOund). The maximization of ELBO, which is equivalent to minimizing Eq. (2.45), can be written as,

$$\Omega^* = \arg \max_{\Omega} \mathbb{E}_q[p(\phi, \mathbf{z}, \mathbf{w}|\Omega)] - \mathbb{E}_q[q(\phi, \mathbf{z}|\Omega)] \quad (2.46)$$

where the expectation  $\mathbb{E}_q$  are with respect to the mean-field variational family  $q(\phi, \mathbf{z}|\Omega)$ .

Moreover, if the complete conditionals in the LVM and the mean-field distributions are all in the Exponential family, one can derive closed-form updates for Eq. (2.46). For the conditionally conjugate LVM in Fig. 2.2, the updates for the variational parameters are the following:

$$\psi_n^* = \mathbb{E}_q[\eta_l(x_n, \phi)] \quad (2.47)$$

$$\lambda^* = \mathbb{E}_q[\eta_g(\mathbf{z}, \mathbf{x})] \quad (2.48)$$

where  $\eta_l(x_n, \phi)$  and  $\eta_g(\mathbf{z}, \mathbf{x})$  are again the natural parameter vectors corresponding to the complete conditionals described in Eq. (2.42) and Eq. (2.43), respectively.

One then sets up a coordinate ascent algorithm that iteratively updates Eq. (2.48) and the  $N$  local variables in Eq. (2.47) until convergence. Convergence can be easily diagnosed by monitoring the increase in ELBO and stopping the process when the objective plateaus.



Similar to Gibbs Sampling, mean-field variational inference enables to derive simple algorithms that approximate the posterior by updating the variational parameters with expectations on the complete conditionals. This has paved the way to develop software packages, such as Infer.NET (Minka et al., 2012), capable of automatizing variational inference in certain type of graphical models. In contrast to Gibbs sampling, the fact that the optimisation goal is often non-convex and variational methods are deterministic might cause variational algorithms to get trapped in local maxima. To avoid this, multiple random restarts might be used to guarantee a good maxima. Nonetheless, the determinism in these methods has also led to algorithms that are computationally faster than their Gibbs sampling counterparts due to the high cost associated with sampling.

To obtain better approximations of the posterior, one has to necessarily go beyond the fully factorised assumption of mean-field. One way is, for instance, to perform structured mean-field which looks for tractable substructures in the problem that can be analytically solved (Saul and Jordan, 1996). Another way is to use an inference network as the approximating distribution and perform stochastic optimisation (Kingma and Welling, 2014). However, these methods are problem specific and their implementation often leads to more sophisticated and slower algorithms.

### 2.5.3 Importance Sampling (IS)

IS (Importance Sampling) is often used to approximate integrals or sums such as the model evidence in Eq. (2.40). The idea is to draw samples from a proposal distribution  $Q(\theta)$  which is large in the regions of  $\theta$  that both  $p(\mathbf{x}|\theta)$  and  $p(\theta)$  have high probability. Then, the IS estimator approximates the probability by computing a weighted average across these samples,

$$p(\mathbf{x}) \approx \frac{1}{S} \sum_{s=1}^S p(\mathbf{x}|\theta^{(s)}) w(\theta^{(s)}) \text{ where } \theta^{(s)} \sim Q(\theta). \quad (2.49)$$

where the weights  $w(\theta^{(s)}) = \frac{p(\theta^{(s)})}{Q(\theta^{(s)})}$  are the fraction between the prior probability and the proposal. This estimator is proved to be unbiased for any proposal distribution  $Q(\theta)$  that has support everywhere  $p(\theta)$  does. However, the optimal proposal distribution, the one that minimise the estimator variance, is also intractable because it requires to normalise the integrand with the same quantity that we seek to estimate, the model evidence.

To get around this, one can use unnormalised proposal distributions, like  $\tilde{Q}(\theta) = p(\mathbf{x}|\theta)p(\theta)$ , to provide an unbiased estimate of the evidence as follows

$$p(\mathbf{x}) \approx \frac{\sum_{s=1}^S p(\mathbf{x}|\theta^{(s)}) \tilde{w}(\theta^{(s)})}{\sum_{s=1}^S \tilde{w}(\theta^{(s)})} \text{ where } \theta^{(s)} \sim \tilde{Q}(\theta) \quad (2.50)$$

where the weights  $\tilde{w}(\theta^{(s)}) = \frac{p(\theta^{(s)})}{\tilde{Q}(\theta^{(s)})}$ . In fact, by using the unnormalised posterior as proposal  $\tilde{Q}(\theta) = \tilde{p}(\theta|\mathbf{x})$ , one obtains the HM (Harmonic Mean) estimator (Newton and Raftery, 1994),

$$p(\mathbf{x}) \approx \frac{1}{\frac{1}{S} \sum_{s=1}^S \frac{1}{p(\mathbf{x}|\theta^{(s)})}} \text{ where } \theta^{(s)} \sim \tilde{p}(\theta|\mathbf{x}). \quad (2.51)$$

Despite the HM method is unbiased and easy to implement, this estimator is also highly unstable due to the fact that the smallest sampled value dominates the harmonic



mean (Newton and Raftery, 1994). This is attributed to the fact of having to estimate the normalising constant of  $\tilde{Q}$  and hence, one might prefer to use a normalised proposal. In fact, one of the things that this thesis will explore is the use of mean-field distributions that are close to the optimal proposal in terms of KL divergence. In other words, we will explore the family of distributions  $Q(\theta) = \prod_k Q_k(\theta_k)$  such that they minimise the forward and reverse KL,

$$Q_L(\theta) = \arg \min_{Q(\theta)} \text{KL}(Q(\theta), p(\theta|\mathbf{x})) \quad (2.52)$$

$$Q_U(\theta) = \arg \min_{Q(\theta)} \text{KL}(p(\theta|\mathbf{x}), Q(\theta)) \quad (2.53)$$

where  $Q_L(\theta)$ ,  $Q_U(\theta)$  will correspond to different types of solutions, because the asymmetry of this divergence.

## 2.6 Evaluation

We distinguish between two different types of evaluation for probability models: intrinsic and extrinsic evaluation. The former type represents the natural way to evaluate probability models independently of the application at hand. The latter is tied to the application or task and we will pay attention to the task of clustering, which is linked to the approach of event detection discussed later.

### 2.6.1 Intrinsic Evaluation

An intrinsic way to evaluate probability models consist in computing the probability of the unseen data via the posterior predictive distribution. For instance, the posterior predictive for the LVM in Fig. 2.2 can be written as,

$$\begin{aligned} p(\mathbf{x}^*|\mathbf{x}) &= \int p(\mathbf{x}^*|\phi)p(\phi|\mathbf{x}) d\phi \\ &= \int \left( \int p(\mathbf{x}^*, \mathbf{z}^*|\phi) d\mathbf{z}^* \right) p(\phi|\mathbf{x}) d\phi \end{aligned} \quad (2.54)$$

where  $\mathbf{x}^*$  are the unseen observations and  $\mathbf{z}^*$ , their corresponding local latent variables. The first row above integrates out the global variables  $\phi$ , whereas the inner integral in the second collapses the local variables  $\mathbf{z}^*$ .

It is common to evaluate Eq. (2.54) by taking a point estimate of the posterior on the global variables  $\hat{\phi}$  and then compute the inner marginal distribution  $p(\mathbf{x}^*|\hat{\phi})$ . When the inner marginal is not tractable to compute, one can use the IS method discussed earlier to approximate it,

$$p(\mathbf{x}^*|\hat{\phi}) \approx \frac{1}{S} \sum_{s=1}^S p(\mathbf{x}^*|\mathbf{z}^{*(s)}, \hat{\phi}) w(\mathbf{z}^{*(s)}) \text{ where } \mathbf{z}^{*(s)} \sim Q(\mathbf{z}^*). \quad (2.55)$$

However, the high dimensionality of the local variables  $\mathbf{z}^*$  requires more complex estimation methods to approximate this integral. For example, Wallach et al. (2009c) and Buntine

(2009) discuss several estimation methods that approximates this marginal distribution for the LDA model. In this thesis, we will extend some of these methods to a broader class of topic models, referred to as PFA.

Another way to evaluate models intrinsically in the statistical community is known as PPC (Posterior Predictive Checks) (Gelman et al., 1996). This method consists in generating data from the posterior predictive distribution and measure the discrepancy with the observed data. If the discrepancy is high, then the model does not capture well the assumptions of the data. For example, Mimno and Blei (2011) used PPC for LDA to determine which topics violated the assumptions of the model.

## 2.6.2 Extrinsic Evaluation

Probabilistic topic models are often used for exploratory data analysis tasks, such as clustering a collection of documents into distinct groups. For example, a cluster can be assigned to every document in the collection through the topic indicator variable  $z_n$  of a MoU model or through the most likely topic in the topic proportions variable  $\theta_n$  in a LDA model. Clustering consists in grouping together documents or, more generally objects, that are similar while keeping apart those that are not. When a *gold standard* exists for a particular clustering problem, the solution can be evaluated extrinsically against it in terms of its goodness on grouping these similar items together and separating dissimilar ones. Amigó et al. (2009) defined four formal constraints that form the desiderata that a good clustering algorithm should accomplish, and hence, a proper evaluation metric must be able to validate. Next, we review these formal constraint and two families of extrinsic metrics that satisfy some/all of them.

The four formal constraints for a clustering metric specified in (Amigó et al., 2009) are:

1. **Cluster homogeneity** states that a cluster must not mix objects from different categories.
2. **Cluster completeness** imposes that a cluster must not split object from the same category into different clusters.
3. **Rag bag** establishes that is better to have one cluster with unclassified objects than to distribute unclassified objects among “clean” clusters.
4. **Cluster size vs. quantity** prefers a small error in a big cluster than a large number of small errors in small clusters.

In what follows we present two families of metrics: set matching and BCubed.

### 2.6.2.1 Set Matching Metrics

The set matching family is composed of Purity, Inverse Purity and their harmonic mean known as F-measure. These metrics are equivalent to the Precision and Recall concepts from Information Retrieval and they are the most popular metrics for evaluating clustering.

Purity is defined as the weighted average across all clusters  $C_i$  of the category that has maximum precision w.r.t this cluster. That is to say,

$$\text{Purity} = \sum_i \frac{|C_i|}{N} \max_j \text{Precision}(C_i, L_j) \quad (2.56)$$

where the weight is the proportion of items in each cluster and precision is the proportion of elements in cluster  $C_i$  that are labelled as  $L_j$ . Mathematically, precision is expressed as,

$$\text{Precision}(C_i, L_j) = \frac{|C_i \cap L_j|}{|C_i|}. \quad (2.57)$$

Higher purity figures indicate that items in a cluster tend to be from the same category, whereas lower values indicates the contrary. Given that the number of clusters is not fixed, purity can be trivially maximised to 1 by placing each item into a different cluster, but it is minimum when all items are grouped into a single cluster.

To compensate for this trivial solution, inverse purity is introduced. Inverse purity is the weighted average across categories  $L_i$  of the cluster that has maximum precision w.r.t. this category. Formally,

$$\text{Inv. Purity} = \sum_i \frac{|L_i|}{N} \max_j \text{Precision}(L_i, C_j) \quad (2.58)$$

where the weight is now the proportion of items in each category and precision, the proportion of elements in category  $L_i$  that are clustered as  $C_j$ . Precision is also defined as Recall but with labels and clusters swapped. That is,

$$\text{Recall}(L, C) = \text{Precision}(C, L). \quad (2.59)$$

Higher Inverse Purity figures indicates that items from a category are grouped into the same cluster, and lower Inverse Purity means the opposite. Hence, Inverse Purity is trivially maximised to 1 when all items are grouped into a single cluster, but it is minimum when each item belongs to a different cluster.

[Van Rijsbergen \(1974\)](#) took the harmonic mean of both Precision and Recall and propose the F-measure or  $F_1$  score. This score is often used for binary classification as a measure of the retrieval accuracy which gives equal importance to Precision and Recall. Weighted versions of this score also exist. Later, [Steinbach et al. \(2000\)](#) also used this score for clustering to compensate for the trivial solutions that Purity and Inverse Purity have. Mathematically, the F-measure for clustering is expressed as,

$$F = \sum_i \frac{|L_i|}{N} \max_j F(L_i, C_j) \quad (2.60)$$

where,

$$F(L_i, C_j) = 2 \cdot \frac{\text{Recall}(L_i, C_j) \text{Precision}(L_i, C_j)}{\text{Recall}(L_i, C_j) + \text{Precision}(L_i, C_j)}. \quad (2.61)$$

Although Purity satisfies the cluster homogeneity constraint presented above, none of the other metrics does as shown in ([Amigó et al., 2009](#)). Besides, none of the three metrics is sensible to the cluster completeness and Rag Bag properties. And only inverse purity and F-measure satisfy the fourth constraint. The reason why these metrics have counterexamples for each of these properties is due to the fact that they only consider the cluster/category that has maximum precision, recall or F-measure, but disregard the others.

### 2.6.2.2 BCubed Metrics

The BCubed family (Bagga and Baldwin, 1998) calculates the Precision and Recall associated to each item to address the issues of the set matching family in the previous section. As shown in (Amigó et al., 2009), BCubed metrics together satisfy all four constraints previously stated, being the only metric to account for the Rag Bag property.

BCubed family is also composed of the BCubed Precision, Recall and F-measure but, in contrast to set matching metrics, they are defined at the item level. Their definition can be done through the notion of correctness between a pair of objects  $o$  and  $o'$ ,

$$\text{correctness}(o, o') = \begin{cases} 1 & L(o) = L(o') \iff C(o) = C(o') \\ 0 & \text{otherwise} \end{cases} \quad (2.62)$$

where  $L(p)$  is the label of object  $o$  and  $C(o)$ , its cluster assignment. This means that two points are considered to be correctly related whenever they are from the same category, i.e.  $L(o) = L(o')$ , and are grouped into the same cluster, i.e.  $C(o) = C(o')$ .

Then, the BCubed Precision and Recall are defined as the averages over all objects. For each object, precision is computed as the proportion of objects in its cluster that are correctly related (from the same category) and recall is the proportion of objects in its category that are correctly related (from the same cluster). That is,

$$\text{BCubed Precision} = \text{Avg}_o[\text{Avg}_{o'|C(o)=C(o')}[\text{correctness}(o, o')]] \quad (2.63)$$

$$\text{BCubed Recall} = \text{Avg}_o[\text{Avg}_{o'|L(o)=L(o')}[\text{correctness}(o, o')]]. \quad (2.64)$$

As shown by Amigó et al. (2009), BCubed Precision satisfies constraints 1 and 3, while BCubed Recall covers constraints 2 and 4. Besides one can combine both metrics through Van Rijsbergen's F-measure in Eq. (2.61) and obtain a metric that covers all four restrictions. We note, however, that the computation of the BCubed metrics grows quadratically with the number of objects, whereas set matching metrics are in the worst case linear.

# Part I

## Event Detection Task



# 3

## Event Detection in Twitter

*“We do not remember days, we remember moments”*

Cesare PAVESE, 1940

Capdevila, J., Cerquides, J., and Torres, J. (2017b). Event detection in location-based social networks. In *Data Science and Big Data: An Environment of Computational Intelligence*, pages 161–186. Springer

Social networks have been attracting the interest of both academia and industry because they do not only serve as communication platforms, but they also offer numerous opportunities to study and monetize human behaviour. For instance, [Borge-Holthoefer et al. \(2011\)](#) studied the dynamics of the anti-austerity movement that took place in Spain during the spring of 2011 through the analysis of the tweet messages exchanged in Twitter during days prior and after the central day, the 15th of May. Similarly, [Kim et al. \(2013\)](#) demonstrated that monitoring the spread of an epidemic influenza in populations could be done faster from tweet messages than with current practices, which used telephone triage calls, over-the-counter medication sales, among others. In short, social networks have enabled the computerization of social sciences and hence, the advent of new tools to address important questions in this field.

Twitter is one of the most popular social networks and micro-blogging sites with more than 330 millions monthly active users worldwide in 2018<sup>1</sup>. In this network, users post tweet messages in response to the question *What’s happening?*. These messages are composed of 280-character-long (140 at the time of this work) texts that also contain a large amount of

---

<sup>1</sup><https://www.statista.com/statistics/274565/monthly-active-international-twitter-users/>  
[Access: 02/09/2018]

contextual data, such as user name, posting time, geographical localization, among much more metadata. The rise in popularity of Twitter has been often attributed to the faster speed to publish breaking news than to traditional channels. For example, the Mumbai terrorist attacks (Stelter and Cohen, 2008) or the Osama Bin Laden raid (Newman, 2011) were first reported in Twitter by eyewitnesses. From a technical point of view, the ease of access to the public data through a friendly API (Application Programming Interface) has also boosted its popularity among the developer community and social computing researchers, despite the compliance of Twitter legal terms which do not allow you to share it even for non-commercial uses.

Along this line, an event happening in a specific location (such as a demonstration, a music concert, an accident or a street fight), a.k.a., at the “local level” (Lee, 2012), is likely to be reported on Twitter through geo-referenced tweet messages posted by eyewitness users. Identifying and summarising these local events, their temporal and spatial extent, their social composition, etc. has become an interesting research problem with a broad range of applications (Panagiotou et al., 2016). For instance, a city council might want to know about events that have happened in its city during the past week, month or year in order to assess and plan future events. Spreading a team of pollsters might be too costly and still incapable of identifying certain types of events (e.g. unscheduled events) or dimensions (e.g. social relationships). Through data flowing from social networks via their public APIs, pattern detection techniques that automatically identify events and geographic information systems to explore and visualize them, one can envision end-to-end systems capable of providing actionable insights for the city authorities. See, for instance, the European funded project Insight<sup>2</sup> to develop methods and systems that enable to improve the emergency responses in cities and countries or the commercial products developed by Event Registry<sup>3</sup> to monitor and explore worldwide events.

The problem of event detection has been extensively studied in traditional media channels (Allan et al., 1998) as well as in surveillance systems (Wong and Neill, 2009). However, each field took a slightly different definition of the problem and hence have given rise to distinct detection methods. Moreover, the type of data in media channels (e.g. images, text, videos) is different from that in surveillance systems (e.g. sensor readings, spatio-temporal) and also from that in social networks like Twitter (e.g. high-dimensional and multimodal data). These two approaches have influenced the problem in social networks and, as a result, the existing literature contains a mix of methods depending on whether they focus on the spatio-temporal features of tweets or on the dynamics of text over time. In this chapter, we will first present a definition for the broad task of event detection in social networks. Then, we will review the literature for the problem of local event detection and we will classify the existing methods according to their influences.

In contrast to event detection in traditional news media, the problem definition will expose that not all postings in a social network can be categorized as event-related since most might have other intentions. For example, a global event about the release of the final episode of the TV show Game of Thrones has to be ignored by a detection method of local events. To deal with this, we will review different approaches in the literature to identify event-related publications, and we will assert the need to explicitly model this identification process in any detection method. In particular, we will show that methods developed by

---

<sup>2</sup><http://www.insight-ict.eu/> [Access: 02/09/2018]

<sup>3</sup><http://eventregistry.org/> [Access: 02/09/2018]



the spatial statistics community might be more suitable to this definition because they also consider that an event is reflected as an anomalous pattern in the data, and hence their methods are based on anomaly detection.

Moreover, Twitter poses a set of special features that make the task of event detection challenging and different from that in traditional media channels and in surveillance systems. Text messages addressing *What's happening?* are sometimes too short and too informal to understand their meaning for someone without contextual information. Besides this, brevity is also one of the main limitations of probabilistic topic models (Blei, 2012) to learn good semantic representations, as discussed in Chapter 2. Therefore, event detection techniques for Twitter need to address the tweet shortness in order to identify semantically meaningful events. Fortunately, the abundance of contextual information offers a unique opportunity to either pool tweet messages together or build hierarchical probability models that leverage on the context.

In this chapter, we review in Section 3.1 the problem of event detection in social networks, and more specifically in Twitter. In Section 3.2, we provide an overview of the related work in this field. Finally, we present in Section 3.3 a publicly available data set for the task of local event detection in Twitter, which will be used in the subsequent chapters.

### 3.1 Event Detection: Problem Definition

Event detection was first studied under the project Topic Detection and Tracking (TDT) (Allan et al., 1998) for traditional media channels. The project was focused on (1) *segmenting* a stream of data into distinct stories, (2) *detecting* events from this corpus of stories and (3) *tracking* the evolution of events as well as their association to stories. The project also distinguished between two types of detection, either RED (Retrospective Event Detection) or NED (New Event Detection). While the former sub-task consisted in retrieving all events from the stories basically through clustering, the latter was more about processing stories sequentially and associating these to new events in an on-line manner. In this dissertation, we focus on the RED problem, which we also refer to as event discovery to emphasize the exploratory nature of this task.

The problem of event detection is also studied in other fields beyond news media. For instance, the early detection of events like disease outbreaks or terrorists in video surveillance systems are occurrences of critical importance for public safety (Weng and Lee, 2011). Event detection in sensor networks for surveillance considers that events are groups of anomalous observations hidden in the data. Therefore, the goal of event detection is not simply to cluster observations, but it is also to uncover the anomaly observations that compose these events.

Event detection in social networks like Twitter (Atefeh and Khreich, 2015) inherits a little bit of both worlds. On the one hand, social networks are media channels in which users publish stories about relevant events, but these channels also serve as communication platforms with tones of non-event messages. On the other hand, social networks can also be seen as sensor networks in which users, acting as sensors, report daily occurrences, but the reported information is in the form of noisy and unstructured text. The different views of event detection in social networks have led to a myriad of methods, each tackling the problem in a different way.

An important step on unifying the different views on the problem was proposed in [Paniotou et al. \(2016\)](#), where the authors provided a formal definition for the problem that takes into account previous attempts and that accommodates the distinct types of events: planned, unplanned, breaking news, local or entity related. Their definition is based on the observation that an *event* is the cause of an increase in the number of *actions* performed by *accounts*. These *actions* can be any type of interaction with *content* (e.g. like, retweet) or with other *accounts* (e.g. follow, unfollow).

**Definition 3.1.1.** Account( $p$ ): An agent that can participate (i.e. perform actions) in a social network after following a registration procedure.

Therefore, *accounts* can be individuals, organizations or computer bots. In some cases, one might want to exclude bots or organizations if they do not contribute to the type of event that is being explored. For instance, in local event detection bots are often excluded because they are unlikely to contribute to a real-world event.

**Definition 3.1.2.** Content Object ( $c$ ): A textual or binary object that is published or shared via the social network (e.g. text, image, video).

*Content objects* are published by *accounts*. A *content object* might also be enriched with extra information, such as tweets which sometimes combines text and images or video. Content might also be geo-located, meaning that it can be geographically referenced through exact coordinates of the GPS system or associated with a specific place (e.g. Piccadilly Circus).

**Definition 3.1.3.** Action ( $a$ ): Depending on the social network, an action,  $a$ , can be either: (i) a post of new content (e.g. a new tweet) (ii) an interaction with another profile (e.g. a new follower, a friend request, etc.) (iii) an interaction with another user's content (e.g. a retweet, or a "like").

Thus, *actions* are performed by accounts either through the publishing of *Content objects* or through the interactions with other *account's* profile. Finally, an *event* is defined as the cause of the increase of these *actions*.

**Definition 3.1.4.** Event( $e$ ): In the context of social networks, (significant) event  $e$  is something that causes (a large number of ) actions in the social network.

An *event* is significant if it is associated with (or causes) an increase of the actions, but we also note that a decrease of actions could be an interesting type of event to consider (e.g. a terrorist attack might cause a decrease of actions in the area under attack).

Finally, the task of event detection consists in identifying and characterizing groups of *actions* that have been caused by the *event*. Therefore, it is clear that the identification sub-task consists in (i) distinguishing event-related actions from those that are not, (ii) grouping these actions into events. Besides the characterization sub-task seeks to summarise the events through the features of these actions.

**Definition 3.1.5.** Event detection in a social network: Given a stream of actions  $A_n$  of the social network  $n$ , identify all tuples  $E = \{e_1, \dots, e_M\}$ , such that  $M$  is the number of events and  $e_i = \langle R(e_i), A^{e_i}, T_A^e, \text{loc}_{e_i}, I^{e_i} \rangle$  is the set that contains some of the following information:

- (a) the (textual) representation of the event  $R(e)$ ,
- (b) a set of actions that related to this event  $A^e \subset A_n$ ,
- (c) a temporal definition of the set of actions  

$$T_A^e = [t(A^e, \text{start}), t(A^e, \text{end})]$$
- (d) a location  $\text{loc}_e$  that is correlated with the event,
- (e) the involved accounts  $I^e$ .

To perform the identification of events, one might proceed in a supervised or unsupervised manner. When a labelled data set exists or the events are topic-specific, supervised machine learning techniques might be more effective for the task. However, labelling a data set is costly and one does not always know which types of event is looking for, such as for unspecified events. In such cases, unsupervised machine learning techniques can be more flexible because they are not constrained to a particular type of event.

In our case, we focus on detecting unspecified local events from geo-located tweets (i.e. *content objects*). Therefore, we are interested in identifying the cause (i.e. *events*) behind an increase of the tweeting activity (i.e. *actions*) in a region. *Accounts* can be either individual or organizations, but bots should be excluded since they can induce virtual events. Moreover, we are also interested in summarising events in terms of their spatio-temporal, textual and user features. In the next section, we review the literature of detection methods that address this very same problem.

## 3.2 Related Work

The multiple definitions of the problem have led to a vast literature of event detection methods in social networks (Panagiotou et al., 2016), as well as in Twitter (Atefeh and Khreich, 2015). In this section, we present a condensed revision of the existing work that is most related to our particular problem of discovering unspecified local events from tweets. Therefore, we exclude those techniques that are thought to detect specified events (e.g. earthquake trackers (Sakaki et al., 2010), crime and disaster detectors (Li et al., 2012) or traffic monitoring systems (D’Andrea et al., 2015)) or non-local events (e.g. Event Detection With Clustering of Wavelet-based Signals (Weng and Lee, 2011) or the trending topic detectors (Becker et al., 2011)). We redirect the reader to surveys (Atefeh and Khreich, 2015; Panagiotou et al., 2016) for an exhaustive revision of these other methods.

We classify the event detection methods depending on which technique they are based: anomaly detection or clustering. Besides, methods that are clustering-based can be further split into supervised and unsupervised in reference to the level of supervision to identify the event-related clusters.

### 3.2.1 Anomaly-based Methods

The development of anomaly-based methods for event detection in social networks was deeply influenced by the field of spatial statistics. This community had already developed sophisticated detection techniques, such as the spatial scan statistic (Kulldorff, 1997), when

social networks appeared. Later on, with the advent of sensor networks, these techniques evolved to deal with more complex features and patterns. For instance, [Kulldorff et al. \(2005\)](#) extended ([Kulldorff, 1997](#)) to enable the detection of spatio-temporal patterns and [Wong et al. \(2005\)](#) proposed an algorithm called WSARE to detect disease outbreaks from multivariate time series, such as health records.

With the appearance of social networks, these techniques start to be applied in this new domain. First methods like [Lee and Sumiya \(2010\)](#) split the region into sub-areas to monitor the anomalous patterns inside these sub-areas. Sub-areas were created by partitioning the space with a K-means clustering on the geo-located tweets and creating a Voronoi map from the clusters' center. For each Voronoi cell, they defined the normality or usual pattern in order to detect the anomalous patterns on it through the quartile ranges in a boxplot. Similarly, [Garcia-Gasulla et al. \(2014\)](#) aggregated tweets into cells that expand 15 minutes along time and regions of  $0.55 \text{ km}^2$  to associate events to cells that follow a particular anomalous pattern. The anomaly is defined by means of a specific rule that also makes use of the deviations in the interquartile range. [Krumm and Horvitz \(2015\)](#) proposed a similar approach that, in addition to partition space uniformly, also explores different spatial and temporal resolutions to detect events that spread over multiple cells. A statistically sound approach was introduced by [Cheng and Wicks \(2014\)](#), who used the STSS (Spatial Scan Statistic) developed in ([Kulldorff et al., 2005](#)) to detect unspecified events from the spatio-temporal features of a tweet.

All methods above focus on finding unspecified events from the spatio-temporal features of geo-located tweets, hence disregarding the textual information in them. Despite showing their effectiveness for certain types of events that significantly change the tweeting activity, distinguishing fine-grained events that overlap spatio-temporally with other occurrences necessarily requires to take text into account. Lately, the spatial statistics community has been researching ways to develop semantic scan statistics to detect events in geo-located text. For example, ([Maurya et al., 2016](#)) explores the combination of topic models ([Blei, 2012](#)) and scan statistics.

### 3.2.2 Clustering-based Methods

The development of clustering-based methods is closely related to the TDT (Topic Detection and Tracking) project ([Allan et al., 1998](#)) in traditional media channels. As stated earlier, event detection in the TDT project was more about associating stories to events than finding anomalous patterns in the text stream. Therefore, most methods were based on clustering documents (e.g. through a probability model ([Li et al., 2005](#))) or their bursty features (e.g. through a Gaussian mixture ([He et al., 2007](#))) to associate events to documents or to group features, respectively. However, these methods could not be directly applied to social networks and Twitter, because most social content is not event-related. Therefore, the different strategies to identify which clusters were event-related have induced two different types of identification methods: supervised and unsupervised.

#### 3.2.2.1 Supervised Cluster Identification

As the name suggests, the identification of event-related clusters is done in a supervised manner. This means that a labelled data set is required to train a classifier to distinguish between classes.

Boettcher and Lee (2012) proposed EventRadar, a scheme to detect local events by first clustering tweets with DBSCAN (Density-based Spatial Clustering of Applications with Noise) (Ester et al., 1996) and then applying a logistic regression to identify which clusters are event-related. They showed that EventRadar outperforms Jasmine (Watanabe et al., 2011), an unsupervised method that identified event-related groups merely based on the word co-occurrence in tweets from the same group. Similarly, Walther and Kaiser (2013) presented a system that clusters tweets as per their spatio-temporal characteristics following a simple rule-based clustering algorithm and then scores each candidate cluster via a classifier that uses textual and non-textual features. Authors reported that the presence of different users in the cluster and the use of the same words in the clustered tweets were the features that had the biggest impact on the performance of the classifier.

Despite the good performances of these methods, the cost of tagging a data set as well as maintaining it updated for new events motivated the study of fully unsupervised identification approaches.

### 3.2.2.2 Unsupervised Cluster Identification

In contrast, methods that perform unsupervised cluster identification use a scoring function to determine which clusters are event-related and which are not. In this group, one can find *feature-pivot* and *document-pivot* methods. While the former is based on clustering words, the latter focuses on clustering documents.

Chen and Roy (2009) presented an event detection method for Flickr photos which exploits their spatio-temporal features and textual annotations and disregards the picture. The method is considered to be *feature-pivot* since it filters out the spatio-temporal noisy terms in the annotations through wavelet analysis and, then, clusters them to form events. Clustering is performed through the DBSCAN (Ester et al., 1996) algorithm with a tailored distance metric that takes into account the semantic similarity of terms in the cluster as well as their spatial distance. Finally, photos are associated to events by finding those with event-related terms in their annotations and checking that their spatio-temporal features also match those of an event.

In Abdelhaq et al. (2013), authors presented EvenTweet, a system to detect local events from tweets. It also employs a *feature-pivot* method that clusters keywords as per their spatial signature, where the spatial signature is the spatial distribution of a keyword. This spatial distribution is defined per cell in terms of the normalised usage of the keyword in each cell. Finally, clusters whose keywords have high burstiness, i.e. words that are suddenly used, but low spatial entropy are identified as event-related.

Another method that uses density-based clustering but is *document-pivot* based was proposed by Lee (2012). The solution first clusters tweets with the incremental DBSCAN algorithm as per their textual and temporal features. In fact, authors tuned up the clustering metrics with a dynamic term weighting scheme that accounts for word burstiness. Then, event-related clusters are mapped into locations by maximizing the probability of the event over all possible locations. Events are considered global if the maximum probability is below a certain threshold and hence, not mapped to a specific location.

In Singh (2015), authors also proposed an extension of DBSCAN algorithm to cluster tweets as per their spatio-temporal and textual features. Textual features were represented through a term vector model and tweet messages compared in terms of the cosine similarity. The resulting clusters were associated to real world events like football matches and



Oktoberfest.

Similarly, [Zhang et al. \(2016a\)](#) proposed GEOBURST, a *document-pivot* method to detect local events that was shown to outperform the *feature-pivot* methods introduced in [Chen and Roy \(2009\)](#) and in [Abdelhaq et al. \(2013\)](#). This detector first clusters tweets as per their spatial and semantic information and then identifies event-related clusters as per their spatio-temporal burstiness.

A fully probabilistic approach to the event detection problem was proposed by [McInerney and Blei \(2014\)](#). Their probability model is a sort of mixture model commonly used for clustering in which events are the latent assignments of tweets and the observed tweets are generated from the mixture components over the spatio-temporal and textual features. On the textual dimension, a topic model ([Blei, 2012](#)) is used to summarise the semantic information of tweets. To learn newsworthy events, an external news data set is also used in order to transfer their semantic information to the probability model. However, this model does not propose any scoring function to distinguish between event and non-event tweets, and hence their identification is performed by a qualitative exploration of topics.

In summary, we have covered two types of methods for unspecified local event detection. The first was inspired by techniques developed in the spatial statistics field and the second was more influenced by methods in the TDT project. Whereas there are fewer attempts to include the textual features of tweets in anomaly-based methods, clustering-based methods have historically been built taking into account this unstructured type of information, either through *document-pivot* or *feature-pivot* techniques. Within clustering-based methods, we have distinguished between those that supervise the identification of event-related clusters and those that do not. The former requires a tagged data set with event and non-event categories, which might hamper the development of real-time systems in fast-paced environments. The latter requires the use of a scoring function or metric that takes into account the characteristics of events. In this latter group, we have seen proposals that first apply clustering and then identify event-related clusters, but also schemes that use clustering methods that intrinsically can deal with non-event observations like DBSCAN.

### 3.3 “La Mercè”: a Data Set for Local Event Detection

One of the issues that slowed down the progress of event detection in social networks is the lack of publicly available data sets. Although Twitter allows one to crawl tweets and other social content through its API, the sharing of this data is forbidden even for academic purposes, as stated in Twitter’s legal terms. A workaround consists in publishing the list of tweet IDs that compose the data set and recovering the tweets by querying the Twitter API with these specific IDs. However, it might still not be possible to recover the whole data set because some tweets might have been deleted by users or some users might have decided to switch their profile into private mode, compromising the reproducibility of experiments.

The TREC (Text REtrieval Conference) organized a Microblog Track from 2011 to 2015, in which a data set of approximately 16 million tweets was released yearly by publishing a list of tweet IDs ([McCreadie et al., 2012](#)). Moreover, the data set contained the judgement on 50,324 tweets such that 2,965 were considered relevant to one of the 49 selected topics. However, the data set was designed for ad-hoc retrieval, and hence, the topics and relevance judgements are unsuitable for event detection.

To address this, McMinn et al. (2013) proposed to build a large-scale corpus for evaluating event detection. In contrast to the TREC corpus, they identified tweets as per their definition of event which differs from ours. For them, an event was a significant thing that had happened in a specific place and time and it was significant if it had been discussed in the media. Thus, their definition did not take into account the activity in the social network and it relied on external information. They crawled 120 million tweets in English from Twitter during 28 consecutive days and they grouped these tweets through existing event detection and information retrieval techniques that leverage on the event information published in the Wikipedia Current Events Portal<sup>4</sup>. Finally, they used crowdsourced workers to decide whether grouped tweets were event-related or not. Moreover, events were not necessarily local, but of global interest in order to appear in the media.

To the best of our knowledge there is no public data set composed of a list of tweet IDs for studying the problem of event detection as defined in Section 3.1. As a consequence, we decided to crawl and label our own Twitter corpus in the city of Barcelona during a week full of local events.

### 3.3.1 “La Mercè” Data Set

“La Mercè” are the local festivities of Barcelona that take place yearly during a week in late September. Several types of events, like music concerts, free museums days, fireworks etc. take place in many parts of the city at different times. Most of these events are reported on in social networks by attendees who share pictures and text information of these events. Besides the existence of a public agenda with the scheduled time and place for all events enable the manual identification of all planned events<sup>5</sup>. Therefore, we propose “La Mercè” as a testbed to study the problem of local event detection.

We have gathered a data set of tweets through the Twitter streaming API<sup>6</sup>. In particular, we have established a long standing connection to Twitter public stream which filters all tweets geo-located within the bounding box of Barcelona city. After that, only tweets that were exactly within the boundaries of the city were considered. This connection was established during the local festivities of “La Mercè”, that took place from the 19th to the 25th of September in 2014 and from 18th to 24th of September in 2015.

|                 | #Tweets | #Tagged tweets | #Tagged events |
|-----------------|---------|----------------|----------------|
| “La Mercè” 2014 | 43.572  | 522            | 14             |
| “La Mercè” 2015 | 12.159  | 635            | 15             |

Table 3.1: “La Mercè” local festivities data sets.

Table 3.1 summarises the main statistics for the two hand-crafted data sets of “La Mercè”. We first note that the number of tweets collected in 2015 is less than in 2014. This is because Twitter released new smart-phone apps in April 2015 that enable to attach a location to a tweet (such as a city or place of interest) instead of the precise coordinates<sup>7</sup>; and lots of

<sup>4</sup>[https://en.wikipedia.org/wiki/Portal:Current\\_events](https://en.wikipedia.org/wiki/Portal:Current_events) [Access: 02/09/2018]

<sup>5</sup><http://lameva.barcelona.cat/merce/en/> [Access: 02/09/2018]

<sup>6</sup><http://dev.twitter.com/streaming/overview> [Access: 02/09/2018]

<sup>7</sup><https://support.twitter.com/articles/78525> [Access: 02/09/2018]





| “La Mercè” 2014                      | “La Mercè” 2015                      |
|--------------------------------------|--------------------------------------|
| Food market - day 1 (31) and 2 (22)  | Food market - day 1 (30) and 2 (40)  |
| Wine tasting - day 1 (18) and 2 (49) | Wine tasting - day 1 (20) and 2 (25) |
| Human towers - day 1 (5)             | Human towers - day 1 (24) and 2 (31) |
| Fireworks (34)                       | Fireworks (54)                       |
| Bogatell concerts (50)               | Bogatell concerts (90)               |
| Projections (16)                     | Giants and Bigheads (10)             |
| MACBA concerts (83)                  | Firerun (67)                         |
| Fabrica Damm concerts (39)           | Maria Cristina concerts (31)         |
| Maria Cristina concerts (19)         | OBC Sagrada Familia concert (10)     |
| <i>CaixaForum conference</i> (15)    | <i>Drupal conference</i> (18)        |
| <i>Atypl conference</i> (58)         | <i>Political meeting</i> (17)        |
| <i>Pro-referendum protest</i> (83)   | <i>Barça football game</i> (168)     |

Table 3.2: Labelled events in “La Mercè”.

were related to the event. As shown in Table 3.1, a total of 511 tweets were associated with 14 events in 2014 and 476 tweets, with 15 events in 2015. Table 3.2 shows the list of events that were manually tagged in “La Mercè” 2014 and 2015 and the number of tweets found per event between parenthesis. At the bottom of the table, in *Italics*, we also show three events that were not in “La Mercè” agenda but occur in Barcelona during those days. These events were discovered by manual inspection with QGIS. Therefore, the final list of events is very diverse ranging from cultural or leisure events to political acts or gastronomic tastings.

Both data sets, “La Mercè” 2014 and 2015, are made public<sup>8</sup> as lists of tweet IDs. As explained above, this publishing procedure is in accordance to Twitter’s Terms of Service because we do not distribute tweets, but simply their IDs. Tweets, if not deleted or privatized by their owners, can be easily recovered through the Twitter public API through the *statuses/show/:id* endpoint. We also provide a list of event-related tweets labelled according to the event.

### 3.4 Summary and Conclusion

In this chapter, we have introduced event detection in Twitter, a data mining task that has attracted the interest of academia and industry due to the new challenges and opportunities that this task entails. We have seen that this popularity has also caused multiple definitions of the concept of event in the literature, and hence a myriad of methods and techniques to detect them has been already proposed.

Because of this, we have presented an agreed definition of event as something that causes a large number of actions in the social network, and hence, the task of event detection consists in detecting these abnormal increases of actions and grouping them into event-related clusters. Under this umbrella, we have classified the existing literature for the problem of local event detection into three groups depending on the type of method used.

Moreover, the revision of literature highlights the lack of publicly available Twitter data

<sup>8</sup><https://github.com/jcapde/Twitter-DS> [Access: 02/09/2018]

sets, hampering the advance of the research field. As a result of this, we have crawled and published our own data set composed of geo-located tweets in the city of Barcelona during its local festivities “La Mercè” in 2014 and 2015. Manual tagging of the schedule events during the festivities of “La Mercè” will enable the evaluation of local event detection methods against this ground truth.

# 4

## Tweet-SCAN: a Heuristic Approach

*“An algorithm must be seen to be believed,  
and the best way to learn what an algorithm is all about is to try it.”*

Donald KNUTH, 1969

Capdevila, J., Cerquides, J., Nin, J., and Torres, J. (2015). Tweet-SCAN: An event discovery technique for geo-located tweets. In *Artificial Intelligence Research and Development: Proceedings of the 18th International Conference of the Catalan Association for Artificial Intelligence*, volume 277, page 110. IOS Press

Capdevila, J., Cerquides, J., Nin, J., and Torres, J. (2017a). Tweet-SCAN: An event discovery technique for geo-located tweets. *Pattern Recognition Letters*, 93:58 – 68. Pattern Recognition Techniques in Data Mining

Most event detection methods in the previous chapter relied on text representations of tweets that often consisted of simple vectors of indexed words. These representations might compromise the detection performance if they are not flexible enough to capture complex semantic relationships. A wide range of probabilistic topic models (Blei, 2012) have been proposed to better represent the high-dimensional thematic structure in text in terms of topics. However, as we have discussed in Chapter 1, these topic models usually do not work well in short text such as tweets. Pooling strategies (Hong and Davison, 2010), i.e. aggregations of short documents by some contextual feature, are known to improve the coherence of topics and hence, can be also beneficial for event detection.

In this chapter, we explore the use of topic models for representing the textual features of tweets in an event detection setup. In particular, we extend a well-known spatial clustering algorithm called DBSCAN (Density-based Spatial Clustering of Applications with Noise) (Ester et al., 1996) to also consider the temporal and textual dimensions of

tweets. Because DBSCAN intrinsically deals with noise points, i.e. points which are not densely packed, it has been extensively used for event detection in social networks as a clustering-based method with unsupervised cluster identification (Lee, 2012; Singh, 2015; Zhang et al., 2016a). Although Zhang et al. (2013) have explored different ways to combine the LDA (Latent Dirichlet Allocation) topic model (Blei et al., 2002) with DBSCAN to learn geographically meaningful topics, their integration for event detection purposes remains unexplored.

The proposed solution, called Tweet-SCAN, associates events to sets of density-connected tweets as per their spatio-temporal and textual features. While time and location of tweets are directly represented in the euclidean space, text messages are modelled in terms of topic distributions learned through the non-parametric HDP (Hierarchical Dirichlet Process) model in Section 2.4.3 and messages are compared with a proper distance metric for probability distributions.

The structure of this chapter is as follows. In Section 4.1, we present Tweet-SCAN, an algorithm based on DBSCAN, but better described in terms of its generalised version GDBSCAN (Generalised Density-based Spatial Clustering of Applications with Noise) (Sander et al., 1998). In Section 4.2, we evaluate the event detection performance of Tweet-SCAN in “La Mercè” dataset, presented earlier in Section 3.3. In fact, we aim to understand the role and sensibility of the different Tweet-SCAN parameters as well as the importance of text in discriminating distinct events. We close this chapter in Section 4.3 by summarising the main points of this work .

## 4.1 Tweet-SCAN: a Heuristic Algorithm

We next present Tweet-SCAN, a detection algorithm to discover local and unspecified events from a dataset of geo-located tweets. The algorithm is based on the GDBSCAN (Sander et al., 1998), which generalises DBSCAN (Ester et al., 1996) to cluster spatially extended objects according to their spatial and non-spatial attributes. Therefore, we will associate events to density-connected sets of tweets according to the definition of the GDBSCAN predicates. In the following sections, we define these predicates for the particular task of event detection in Twitter.

### 4.1.1 Events as Density-connected Sets

DBSCAN (Ester et al., 1996) was proposed for uncovering arbitrarily-shaped spatial clusters whose points form a dense or packed group. The notion of density is articulated through the cardinality of the neighbourhood of a point  $o$  within a radius of  $\epsilon$ . In particular, DBSCAN specifies this through two binary predicates:

1.  $\text{NPred}(o, o') \equiv |o - o'| \leq \epsilon$ .
2.  $\text{MinWeight}(o) \equiv |\{o' \in D \mid |o - o'| \leq \epsilon\}| \geq \text{MinPts}$ .

where the first establishes the neighbourhood via the euclidean distance at  $\epsilon$  and the second fixes its cardinality to be greater or equal to  $\text{MinPts}$ .

The fulfilment of both predicates determines if a point  $p$  is directly density-reachable from another point  $q$ , see (left) Fig. 4.1. In this case,  $q$  is considered to be a *core point*

because it satisfies the above predicates, i.e. it has 4 points in its neighbourhood, and  $p$  is a *border point* since it breaks the second predicate, i.e. it has 1 point in its neighbourhood. The notion of being directly reachable is then extended to density-reachable points when  $p$  and  $q$  are far apart, but there is a chain of points in which each pair of consecutive points are directly density-reachable, as in (middle) Fig. 4.1. Finally, it might happen that  $p$  and  $q$  are not density-reachable, but there is a point  $o$  from which they are both density-reachable, that is when  $p$  and  $q$  are said to be density-connected, for example in (right) Fig. 4.1. Note that both points,  $p$  and  $q$ , are *border points*, while  $o$  is a *core point*.

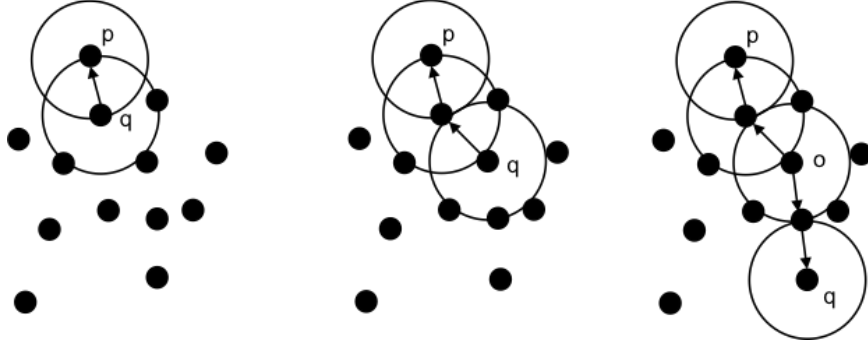


Figure 4.1: DBSCAN definitions: Directly density-reachable (left), density-reachable (middle) and density-connected (right) points for  $\epsilon = \text{radius}$ ,  $\text{MinPts} = 2$ .

Moreover, a cluster in DBSCAN is defined to be a set of density-connected points that contains all possible density-connected points. And, hence, *noise points* are all those points which do not belong to any cluster since they are not density-connected to any.

GDBSCAN (Sander et al., 1998) generalizes DBSCAN by redefining the above predicates to cope with spatially extended objects. For example, the neighbourhood of a set of polygons is defined by the intersect predicate instead of a distance function. It is also the case for a set of points with financial income attributes within a region whose MinWeight predicate is a weighted sum of incomes instead of mere point cardinality, so that clusters become regions with similar income. Therefore, both predicates are generalized as follows:

1.  $\text{NPred}(o, o')$  is binary, reflexive and symmetric.
2.  $\text{MinWeight}(o) \equiv \text{wCard}(\{o' \in D \mid \text{NPred}(o, o')\}) \geq \text{MinCard}$ , where  $\text{wCard}$  assigns a non-negative weight to the whole neighbourhood.

These new predicates enable one to extend the concept of density-connected points to objects and thus generalize density-based clustering to spatially extended objects. Particularly, this extension allows us to formulate the event discovery task from geo-located tweets in terms of GDBSCAN, which we refer as Tweet-SCAN from now on. Within this framework, the resulting clusters will consist of density-connected tweets with respect to its predicates. Because of this, tailored neighbourhood and MinWeight predicates need to be set up in order to associate real world events to density-connected sets of tweets. The following subsections deals with these specifications of the neighbourhood and MinWeight predicates. Besides, we explain how the textual content from a tweet can be modelled in terms of topics.

### 4.1.2 Tweet-SCAN Neighbourhood Predicate

Most event-related tweets are generated throughout the course of an event near the area where it takes place. Although it can happen that some users might be tweeting about an event remotely or even long after it has finished, these tweets are here not considered part of an event. Consequently, it is imperative to find density-connected sets of tweets close in space and time, as well as similar in semantic meaning. We also note that closeness in space is not directly comparable to that in time, nor to that in semantic meaning.

Because of this, Tweet-SCAN is defined to use separate  $\epsilon_1$ ,  $\epsilon_2$ ,  $\epsilon_3$  parameters for space, time and text, respectively. Moreover, specific metrics will be chosen for each dimension given that each feature contains different type of data. The neighbourhood predicate for a point  $o$  in Tweet-SCAN can then be expressed as follows,

$$\text{NPred}(o, o') \equiv |o_1 - o'_1| \leq \epsilon_1 \wedge |o_2 - o'_2| \leq \epsilon_2 \wedge |o_3 - o'_3| \leq \epsilon_3 \quad (4.1)$$

where  $|o_i - o'_i|$  represent here distance functions defined for each dimension, namely space, time and text. The predicate symmetry and reflexivity are guaranteed as long as  $|o_i - o'_i|$  are proper distances. Particularly, we propose to use euclidean distance for the spatial and temporal dimensions given that latitude and longitude coordinates as well as timestamps are real-valued features and straight line distance seems reasonable for this problem. The metric for the textual component will be defined later together with the text model for Tweet-SCAN.

If we scale each metric in the above predicate with its corresponding  $\epsilon_i$  parameter, the predicate then must satisfy that the maximum scaled component is less or equal than 1. Each component being the distance for each separate dimension. Writing this down in terms of the  $\infty$ -norm metric leads to the following expression,

$$\text{NPred}(o, o') \equiv \left\| \frac{|o_1 - o'_1|}{\epsilon_1}, \frac{|o_2 - o'_2|}{\epsilon_2}, \frac{|o_3 - o'_3|}{\epsilon_3} \right\|_{\infty} \leq 1 \quad (4.2)$$

which is equivalent to DBSCAN predicate expressed in terms of the metric scaled by  $\epsilon$  parameter,

$$\text{NPred}(o, o') \equiv \left| \frac{o - o'}{\epsilon} \right| \leq 1 \quad (4.3)$$

Therefore, Tweet-SCAN can be seen as DBSCAN clustering which considers the  $\infty$ -norm of the scaled components as a metric function for the neighbourhood predicate. This result is important since it enables us to determine  $\epsilon_i$  parameter and *MinPts* through heuristics similar to those defined for DBSCAN.

### 4.1.3 Tweet-SCAN MinWeight Predicate

Tweet-SCAN seeks to find clusters of tweets which are generated by a diverse group of users, rather than just a few users. User diversity is imposed to avoid that a few users continuously posting tweets from nearby locations could create an event-related cluster in

Tweet-SCAN. Forcing a certain level of user diversity within a cluster can be achieved through two predicates that must be satisfied at the same time,

$$\text{MinWeight}(o) \equiv |N_{\text{NPred}}(o)| \geq \text{MinPts} \wedge \text{UDiv}(N_{\text{NPred}}(o)) \geq \mu \quad (4.4)$$

where  $N_{\text{NPred}}(o)$  is the set of neighbouring tweets of  $o$  such that  $\{o' \in D \mid \text{NPred}(o, o')\}$  w.r.t. the predicate in Eq. (4.1). The first condition in Eq. (4.4) establishes that neighbouring tweets must have a minimum cardinality  $\text{MinPts}$ , whereas the second condition imposes that the diversity of users in the neighbour must be higher than a threshold  $\mu$ . The diversity is simply defined as the proportion of unique users in  $N_{\text{NPred}}(o)$ .

This combined predicate can be also expressed as one single predicate as in GDBSCAN by,

$$\text{MinWeight}(o) \equiv \min \left( \frac{|N_{\text{NPred}}(o)|}{\text{MinPts}}, \frac{\text{UDiv}(N_{\text{NPred}}(o))}{\mu} \right) \geq 1 \quad (4.5)$$

where  $\text{wCard}()$  function corresponds to the minimum of both quotients and  $\text{MinCard}$  is equal 1.

Note that if we set the user diversity level to 0 in Eq. (4.4), the second condition is always satisfied and the  $\text{MinWeight}$  predicate is simply  $|N_{\text{NPred}}(o)| \geq \text{MinPts}$  which is equivalent to that of DBSCAN. Similarly, if we fix  $\mu$  to 1, we impose that no two tweets in a cluster are tweeted by the same user.

#### 4.1.4 Tweet-SCAN Text Model

Tweet messages are short text fields in which users can type freely their thoughts, experiences or conversations. The fact that users tweet in different languages, argots and styles dramatically increases the size of the vocabulary, making the use of simple term vector models (Salton et al., 1975) not viable. Therefore, we propose to use probabilistic topic models (Blei, 2012), which reduce the dimensionality in a semantically meaningful way. Under this scheme, text messages can be expressed as probability distribution over topics and a meaningful distance metric can be defined over this lower dimension probability space.

In particular, we propose to use the non-parametric topic model introduced in Section 2.4.3, known as HDP. We use the HDP implementation from (Teh et al., 2006b) available here <sup>1</sup> and we use vague informative gamma priors for  $\gamma \sim \text{Gamma}(1, 0.1)$  and  $\alpha \sim \text{Gamma}(1, 1)$  as suggested by authors.

The straightforward use of HDP models on raw tweets does not provide meaningful topics due to the lack of word co-occurrence in short texts like tweets (Hong and Davison, 2010). Because of this, we propose a scheme depicted in Fig. 4.2 which aims to alleviate these shortcomings. First, raw tweets, modelled as Bag of Words, are pre-processed through classical data cleaning techniques from Natural Language Processing (NLP): lowering case, removing numbers and special characters, and stripping white-spaces. Then, processed tweets are pooled together in order to create longer training documents with more word co-occurrences. Next, these training documents feed the HDP model that learns the global topics and the topic distributions for each pooled document in training. Finally, the trained HDP model is used to predict topic distributions for each individual tweet.

---

<sup>1</sup><https://github.com/blei-lab/hdp>

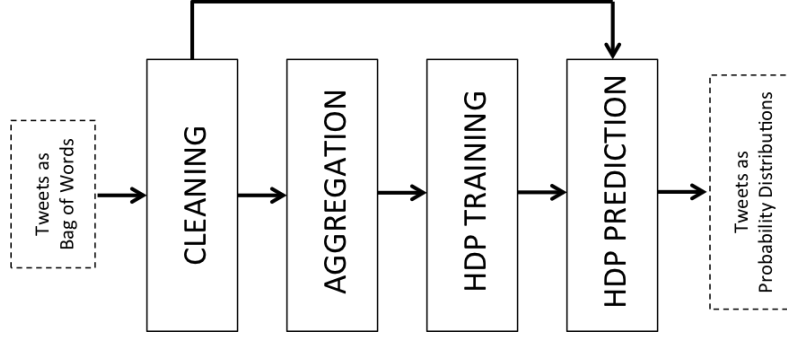


Figure 4.2: Text model scheme.

In this article, we consider two pooling strategies:

- By *hashtags*: it consists in creating a new training document per *hashtag*, which will append all tweets that contains it. Therefore, there will be as many training documents as *hashtags*. One drawback of this aggregation approach is that tweets which do not have hashtags are pooled together, although they might refer to completely different themes. Others (Hong and Davison, 2010) have aggregated tweets by user, but it did not work well for local event detection because users tweeted few times about the same event.
- By top keywords: it consist in first identifying a set of top keywords through the TF-IDF (Term Frequency Inverse Document Frequency) statistic (Salton and Buckley, 1988), and then aggregating by these keywords all tweets that contains them. Thus, there will be as many training documents as top keywords and few tweets will be unassigned as long as we choose a reasonable number of keywords.

Finally, we introduce the JS (Jensen-Shannon) distance to be used for the textual component in Tweet-SCAN neighbourhood predicate. JS is a proper distance metric for probability distributions (Endres and Schindelin, 2003) and hence, appropriate for comparing topic distributions representing different tweet documents. It is defined as,

$$JS(p, q) = \sqrt{\frac{1}{2}D_{KL}(p||m) + \frac{1}{2}D_{KL}(q||m)} \quad (4.6)$$

where  $p$ ,  $q$  and  $m$  are probability distributions and  $D_{KL}(p||m)$  is the KL (Kullback-Leibler) divergence between probability distribution  $p$  and  $m$  written as,

$$D_{KL}(p||m) = \sum_i p(i) \log_2 \frac{p(i)}{m(i)} \quad m = \frac{1}{2}(p + q) \quad (4.7)$$

where  $m$  is the average of both distributions.

In Tweet-SCAN,  $p$  and  $q$  from Eq. (4.6) are two probability distributions over topics which are associated to two tweet messages. Given that Jensen-Shannon distance is defined through base 2 logarithms, JS distance will output a real value within the  $[0, 1]$ . Documents with the similar topic distribution will have a Jensen-Shannon distance close to 0 and those topic distributions which are very far apart, distance will be almost 1.



## 4.2 Experimentation

In this section, we assess Tweet-SCAN for the task of local event detection in “La Mercè”. In particular, we want to show the benefits of using text for event discrimination. We also seek to provide insights on the role of each parameter and its impact to the overall performance.

With the spatio-temporal parameters ( $\epsilon_1, \epsilon_2$ ), the user diversity threshold  $\mu$  and  $MinPts$  set to reasonable values, we first study the impact of the textual parameter ( $\epsilon_3$ ) to the detection performance, in terms of the set matching metrics introduced in Section 2.6.2. Then, we show the importance that the pooling method has for the discrimination of events that overlap in space-time. After that, we analyze the impact that the user diversity threshold  $\mu$  has on the number of detected events and the detection performance. Finally, we explore the detection capabilities for different spatio-temporal and textual configurations.

### 4.2.1 Analyzing the Textual Component

Here, we aim to analyze the textual parameter  $\epsilon_3$  by first fixing the rest of parameters to reasonable values. In particular, we choose to set  $\epsilon_1 = 250m$ ,  $\epsilon_2 = 3600s$  because these values seem to be in accordance to the spatial and temporal extent of the events in “La Mercè” dataset. Besides we also set  $MinPts = 10$  and  $\mu = 0.5$ , thus fixing that the smallest possible event will have 10 tweets and that events are composed of at least a 50 % of unique users. To study  $\epsilon_3$ , we then plot Purity, Inverse Purity and F-measure from Section 2.6.2 as a function of this parameter for “La Mercè” 2014 and 2015, and identify the  $\epsilon_3$  that maximises F-measure.

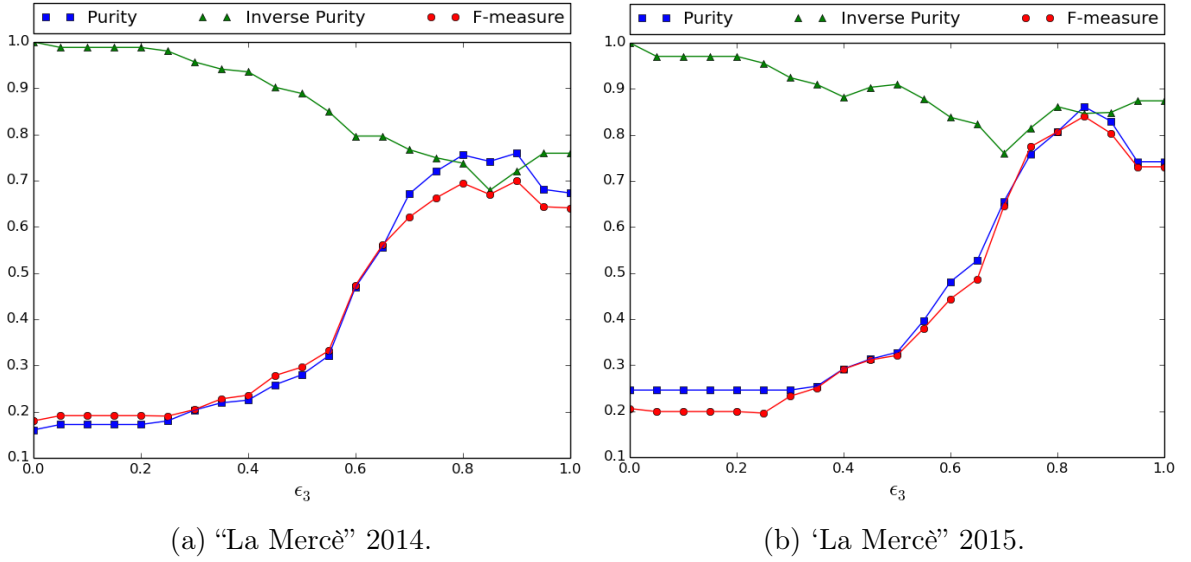


Figure 4.3: Set matching metrics as a function of  $\epsilon_3$  for  $\epsilon_1 = 250m$ ,  $\epsilon_2 = 3600s$ ,  $\mu = 0.5$  and  $MinPts = 10$ .

From Fig. 4.3, it is clear that F-measure is maximum when  $\epsilon_3$  is within the range 0.8-0.9 for both data sets. Given that the maximum is achieved for  $\epsilon_3 < 1$ , this indicates that the textual component improves the clustering performance. If the maximum would have been achieved at  $\epsilon_3 = 1$ , this would mean that the best Tweet-SCAN configuration does not need

text to discriminate between events. Besides, we also observe that Tweet-SCAN performs slightly better in “La Mercè” 2015 than in 2014, what can be explained from the fact that the proportion of tagged tweets in “La Mercè” 2015 is greater than in 2014. The plots also show that the Purity of clusters is optimised for the same range of  $\epsilon_3$ , meaning that the maximum cluster homogeneity is also achieved in this range. In contrast, the Inverse Purity deteriorates with increasing values of  $\epsilon_3$  as expected since  $\epsilon_3 = 0$  corresponds to the trivial solution in which all points are clustered as noise.

| Event                         | “La Mercè” 2014 | “La Mercè” 2015 |
|-------------------------------|-----------------|-----------------|
| Food market - day 1 and 2     | 0.41 and 0.27   | 0.93 and 0.94   |
| Wine tasting - day 1 and 2    | 0.14 and 0.33   | 0.21 and 0.34   |
| Human towers - day 1 and 2    | 0.88 and -      | 0.54 and 0.74   |
| Giants and Bigheads           | -               | 0.34            |
| Fireerun                      | -               | 0.62            |
| Fireworks                     | 0.98            | 0.94            |
| Projections                   | 0.57            |                 |
| MACBA concerts                | 0.45            | -               |
| Fabrica Damm concerts         | 0.97            | -               |
| Maria Cristina concerts       | 0.59            | 0.78            |
| Bogatell concerts             | 0.96            | 0.84            |
| OBC Sagrada Familia concert   | -               | 0.85            |
| <i>CaixaForum conference</i>  | 0.63            | -               |
| <i>Atypl conference</i>       | 0.40            | -               |
| <i>Drupal conference</i>      | -               | 0.88            |
| <i>Pro-referendum protest</i> | 0.94            | -               |
| <i>Political meeting</i>      | -               | 0.27            |
| <i>Barça football game</i>    | -               | 0.99            |

Table 4.1: F-measure per event for  $\epsilon_1 = 250m$ ,  $\epsilon_2 = 3600s$ ,  $\epsilon_3 = 0.8$   $MinPts = 10$ ,  $\mu = 0.5$ .

Table 4.1 shows F-measures for each event tagged in “La Mercè” 2014 and 2015. We observe that the “fireworks” event has been successfully identified both years, while, for instance, the “wine tasting” event has been poorly detected in both. The point is that “fireworks” occurred during the closure of the festivities and in isolation of other events. Something similar happened with “*Barça football game*” which was a huge event that occurred in an area and time with low tweeting activity. On the contrary, “wine tasting” took place near the “food market” event (at Passeig Lluís Companys and Parc de la Ciutadella, respectively) during the same hours (all day events). Since both events were close in space and time and occurred in a region with high tweeting activity, Tweet-SCAN has difficulties to distinguish between them. In fact, we next show that the use of text is in particular beneficial in such situations, where the events might overlap in time and space.

To show this, Fig. 4.4 plots the geo-location of tweets from the “wine tasting” and “food market” events which took place in “La Mercè” 2014 (top) and 2015 (bottom). The left maps in both editions show the geo-location of tweets tagged for these events, where green dots means that tweets belong to the “wine tasting” and orange dots, to the “food market”. Maps in the middle show the clustering results of Tweet-SCAN with the textual component

disabled  $\epsilon_3 = 1$ . This configuration does not only merge both events into the same, but other tweets nearby are also clustered together. Maps on the right-hand side show the clustering results of Tweet-SCAN for a  $\epsilon_3 = 0.8$ , which uncovers both events in “La Mercè” 2014 and one in “La Mercè” 2015. Note that grey dots in the middle and left maps represent tweets clustered as noise. Through this example, we show that text enhance the discrimination between certain types of events but a better textual representation might also boost the detection capabilities of this method.

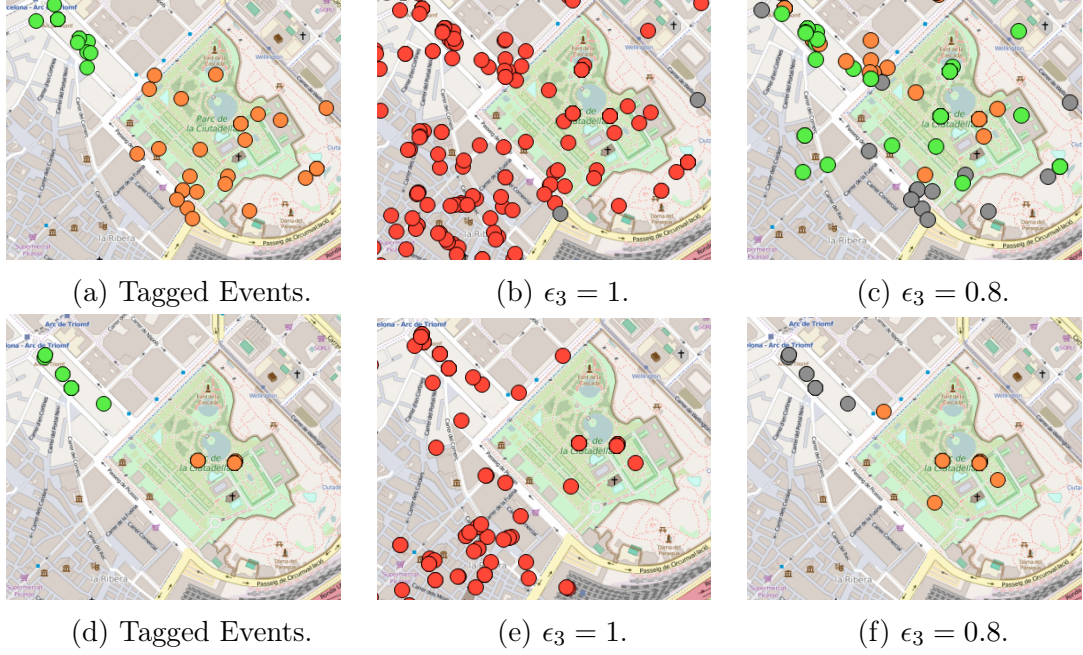


Figure 4.4: Spatial representation of “wine tasting” and “food market” events in “La Mercè” 2014 (top), “La Mercè” 2015 (bottom).

To explore this, we plot in Fig. 4.5a the same Tweet-SCAN results than in Fig. 4.4f. Next to it, we include the histograms of all inter- and intra- class distances in terms of the JS distance of the topic distributions per tweet. Blue bars represent the intra-class distances among tweets in the “food market” event, whereas green bars are the inter-class distances between tweets in the “food market” and tweets in the “wine tasting”. Ideally, one would expect that all intra-class distances were concentrated around 0, whereas inter-class distances, concentrated around 1. However, in reality, if classes are not perfectly separable or the textual representation is not good enough, there exists an overlap of both histograms. As we can see in Fig. 4.5b, the overlapping region for the “food market” and “wine tasting” events is high. Next, we explore whether a different text model would reduce the overlapping and hence improve the detection.

In Fig. 4.6, we study again the discrimination between “food market” and “wine tasting” events in “La Mercè” 2015 by considering now the HDP text model that aggregates tweets by top keyword from Section 4.1.4. As it can be seen in Fig. 4.6a, we can now distinguish between tweets related to the “wine tasting” and those about the “food market” thanks to the larger separation between the histograms of inter- and intra- class distances, see Fig. 4.6b. This preliminary result encourages to develop event-specific topic models with richer textual representations. In Chapter 5, we will develop a method that simultaneously

learns topics and events by grouping tweets per event.

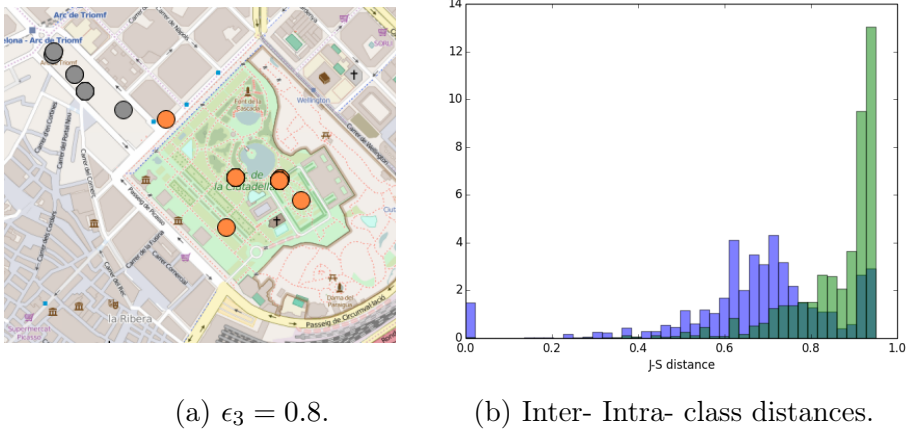


Figure 4.5: Event discrimination in “La Mercè” 2015 when pooling tweets by hashtag.

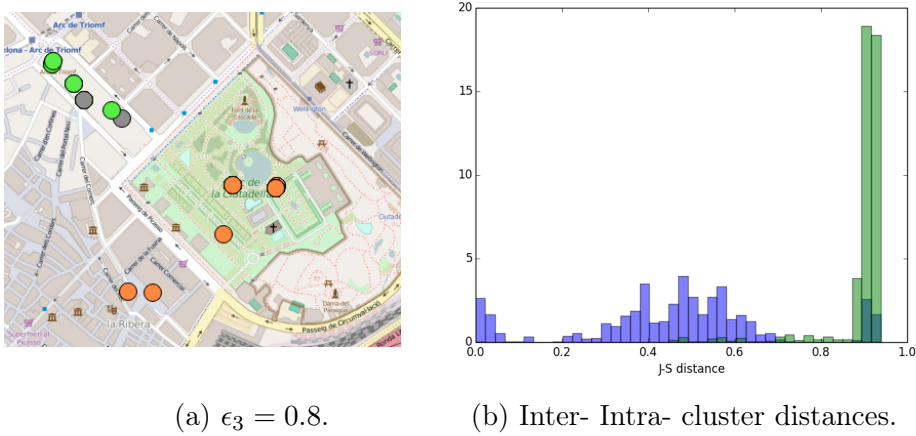


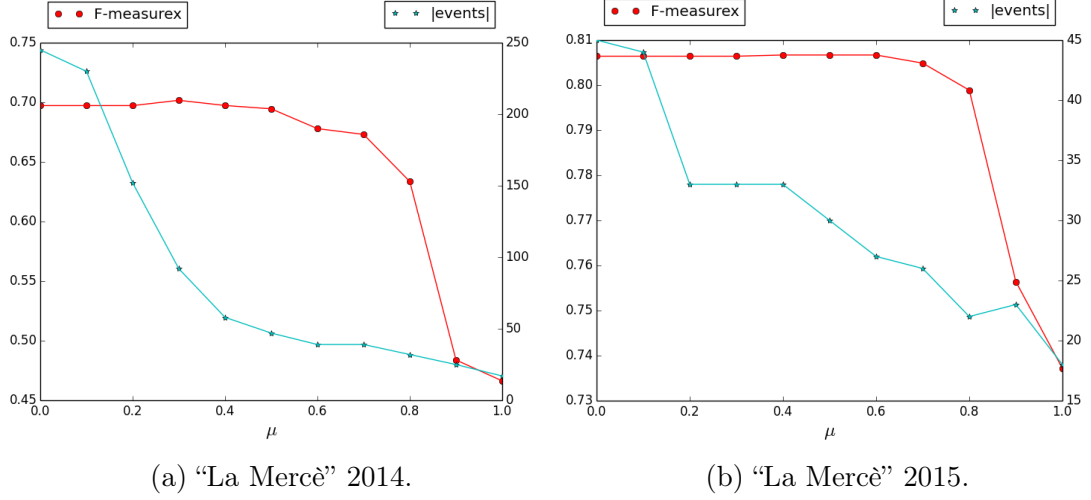
Figure 4.6: Event discrimination in “La Mercè” 2015 when pooling tweets by keyword.

### 4.2.2 Analyzing User Diversity

The role of the user diversity threshold,  $\mu$ , is to guarantee that Tweet-SCAN clusters are composed of a diverse set of users, not a few. By increasing this parameter, we reduce the number of clusters created by a few users or bots actively tweeting from nearby locations about similar topics; but too high values might also impact the detection performance.

In what follows, we examine the effect of different user diversity thresholds  $\mu$  to the clustering results in terms of F-measure and number of events. For that, we use the same spatio-temporal parameters ( $\epsilon_1 = 250m$ ,  $\epsilon_2 = 3600s$ ,  $MinPts = 10$ ) than above and we also set the textual parameter to its optimum value ( $\epsilon_3 = 0.8$ ). We then plot the F-measure as a function of the user diversity threshold  $\mu$ .

Fig. 4.7 plots the F-measure and number of clusters as a function of  $\mu$  for both editions. It is clear that F-measure starts decreasing in both data sets for values of  $\mu$  above 0.6. We observe that a user diversity level of 50% ( $\mu = 0.5$ ) provides a high detection performance and a reasonable number of events ( $\sim 50$  events in “La Mercè” 2014 and  $\sim 30$  in 2015).

Figure 4.7: Tweet-SCAN for different  $\mu$  values.

### 4.2.3 Analysing Spatio-temporal Components

In this section, we analyse the impact of different neighbourhood sizes to the detection performance. Thus, we compute Purity, Inverse Purity and F-measure scores when varying  $\epsilon_1$ ,  $\epsilon_2$  and  $\epsilon_3$ . Fig. 4.8 shows 4 possible configurations of  $\epsilon_1$ ,  $\epsilon_2$  as function of  $\epsilon_3$  for both data sets: "La Mercè" 2014 (top), "La Mercè" 2015 (bottom).

A Tweet-SCAN configuration with smaller neighbourhoods in time and space ( $\epsilon_1 = 250m$ ,  $\epsilon_2 = 1800s$ ) than the ones used in Section 4.2.1, optimises F-measure for  $\epsilon_3 = 1$ . This means that Tweet-SCAN disregards the textual component and it can be explained by the fact that these  $\epsilon_1\epsilon_2$ -neighbourhoods are too restrictive for the tagged events. Besides, the maximum F-measure for this configuration in "La Mercè" 2014 is comparable to that in Section 4.2.1 and hence, is the highest across all configurations in this data set.

For the spatio-temporal values considered in Section 4.2.1 ( $\epsilon_1 = 250m$ ,  $\epsilon_2 = 3600s$ ), we have seen that the optimum value for  $\epsilon_3$  is achieved in the range 0.8-0.9 in both data sets. Now, we can also observe that this spatio-temporal configuration performs much better than others in "La Mercè" 2015 and comparably to the previous configuration in 2014.

If we increase the spatial neighbourhood to  $\epsilon_1 = 500m$ , but we keep the temporal short  $\epsilon_2 = 1800s$ , F-measure lowers in both data sets, and the optimum value is attained for  $\epsilon_3$  within 0.8-0.9 in 2014, and for  $\epsilon_3 = 1$  in 2015. In fact, the curves for "La Mercè" 2015 are very similar to those given by  $\epsilon_1 = 250m$ ,  $\epsilon_2 = 1800s$ , so we think that temporal neighbourhoods of 1800s are too short to achieve a good performance in this data set.

Last, we increase both dimensions to  $\epsilon_1 = 500m$  and to  $\epsilon_2 = 3600s$ . Although the optimum F-measure score is lower than the  $\epsilon_1 = 250m$ ,  $\epsilon_2 = 3600s$  setup in both data sets, we observe that the textual component gains importance, since the larger  $\epsilon_1\epsilon_2$ -neighbourhoods require more textual discrimination to identify meaningful events.

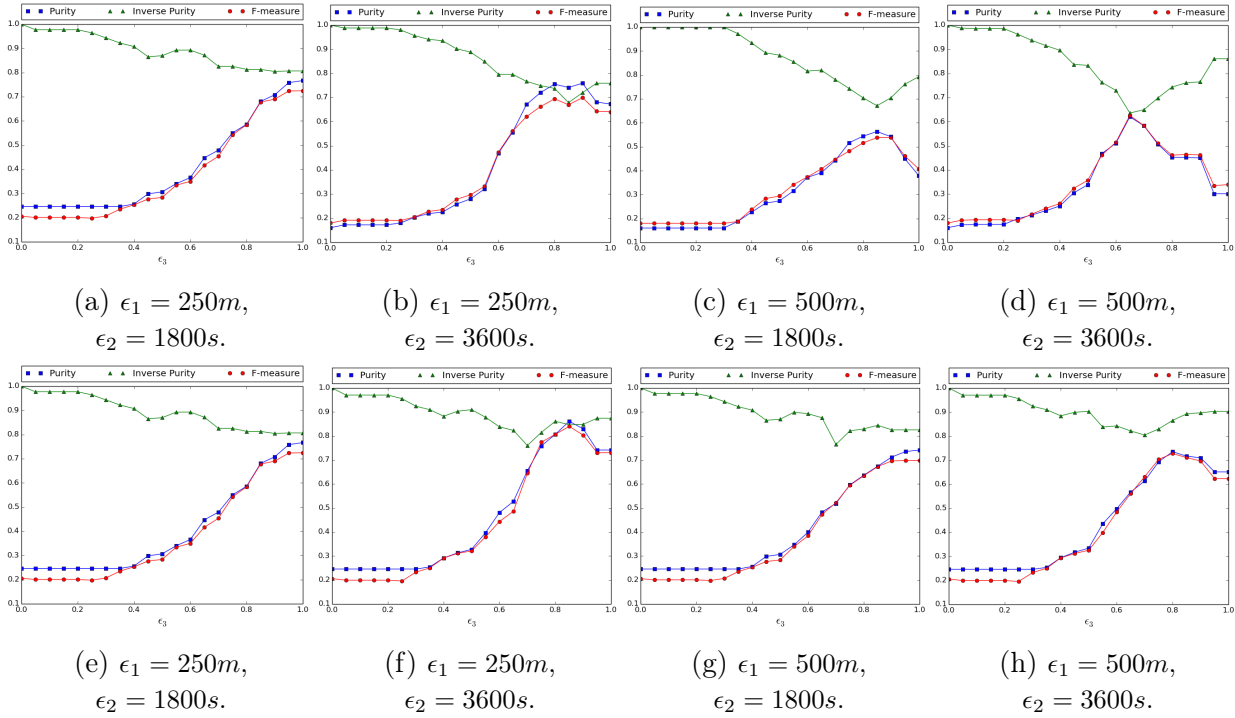


Figure 4.8: Tweet-SCAN for different  $\epsilon_1$ ,  $\epsilon_2$ ,  $\epsilon_3$  and  $MinPts = 10$ ,  $\mu = 0.5$ .  
“La Mercè” 2014 (top), “La Mercè” 2015 (bottom).

### 4.3 Summary and Conclusion

In this chapter, we have presented Tweet-SCAN, an extension of the DBSCAN algorithm that is capable to cluster tweets as per their spatio-temporal, textual and user features. We have re-defined DBSCAN predicates to accommodate for the multimodal information in tweets as well as to perform well in the task of event detection. Furthermore, we have proposed to model text messages through the HDP topic model and a pooling scheme that mitigates the tweet shortness. We have analysed the algorithm performance in “La Mercè” dataset through extrinsic clustering measures. In particular, we have studied the sensitivity that the algorithm performance has to the variation of its parameters.

The results of Tweet-SCAN points out to the benefits of using text, when uncovering events from geo-located tweets, specially for large spatial and temporal neighbourhoods. We have also shown that better text models could help to discriminate overlapping events in space and time, as we have seen for the “wine tasting” and “food market” events. On another level, we have seen that imposing user diversity does not worsen performance if the threshold is carefully set, but it helps to discard clusters of tweets posted by computer bots or conversation groups.

We showed that text can play a big role in discriminating between events, despite the shortness of tweets. Therefore, the development of tailored topic models for event detection in Twitter is another interesting avenue for future research. Specifically, topic models that at the same time pool tweets together (e.g. into events) and learn thematic representation could have a positive impact to the overall detection performance, since they are jointly optimised for the same goal. In the next Chapter, we will present a fully probabilistic

models for event detection that jointly learn event-specific topics.







# 5

## WARBLE: a Probabilistic Approach

*“All models are wrong, but some are useful”*

George E. P. Box, 1976

Capdevila, J., Cerquides, J., and Torres, J. (2016b). Recognizing warblers: a probabilistic model for event detection in twitter. Presented at the Workshop of Anomaly Detection at the International Conference on Machine Learning (ICML)

Capdevila, J., Cerquides, J., and Torres, J. (2018a). Mining urban events from the tweet stream through a probabilistic mixture model. *Data Mining and Knowledge Discovery*, 32(3):764–786

Despite their shortness, tweet messages play an important role in event detection, as we showed in the previous chapter. Their textual content is not only necessary to distinguish between different types of events that overlap in space and time, but it has also shown to improve the Purity of the detected clusters. Previously, we have used two pooling schemes to learn useful topic representations from tweet messages to improve the performance of a detection method. However, the proposed topic model, aggregation scheme and detection method acted as three separate components, which prevented that the components could mutually benefit from each other. In contrast, an integral solution to the problem could enable the topic model to learn topics from tweets pooled at the event level and, similarly, the detection method to benefit from event-specific topic representations.

Therefore, we set ourselves the goal of developing a fully probabilistic model that jointly learns good topic representations and performs well at detecting local events. [McInerney and Blei \(2014\)](#) proposed a probability model that integrates both sub-tasks to discover newsworthy geo-located events from topics learned in an external news data set. However, their model does not perform well at discovering local events that cause an increase of actions

in the social network, mainly because it cannot identify which clusters are event-related. Therefore, an integral solution to local event detection has to explicitly consider that events are anomalous groups of tweets and they must be isolated from other patterns in the data. Moreover, the probabilistic formulation of the solution allows us to consider principled methods for setting the model parameters as well as to deal with partially observed data, e.g. non-located tweets.

In this chapter, we present WARBLE, a probabilistic model and learning scheme that performs topic modelling and event detection in an integrated way, likewise McInerney & Blei’s model. In contrast to them, our model is thought for the task of local event detection in Twitter as defined in Section 3.1 and hence, it addresses three well-known challenges of this problem:

**rarity.** Event-related publications are masked by tones of non-event data such as *memes*, user conversations or *retweet* activities, making it very hard to uncover interesting patterns (Becker et al., 2011).

**text-shortness.** The length limit in the textual component of tweets hampers the application of standard text models which rely on the co-occurrence of words such as traditional topic models (Hong and Davison, 2010).

**variability.** The tweeting activity is not flat along a day (it peaks during late night and falls in early morning, i.e. see Fig. 5.5a), nor over a urban area (it concentrates in the city center and spreads in suburbs, i.e. see Fig. 5.5b) (Li et al., 2013).

WARBLE addresses rarity by grouping non-event tweets together in a separate background component of a heterogeneous mixture model. The spatio-temporal features of this background component are preset through empirical backgrounds learned from geo-located tweets prior to the period of interest. Because of these spatio-temporal empirical backgrounds, the model is able to detect events in varying tweet densities in space and time. For instance, the solution is able to detect events in areas/periods of low tweeting activity (e.g. suburbs, off-peak hours) likewise in those of high activity (e.g. downtown, peak hours). Furthermore, by learning topics and events simultaneously the proposed method is able to exclusively use the tweet stream to obtain useful topic representation, thus dropping the dependence on an external data set or pooling strategies.

Although the Tweet-SCAN algorithm presented in Chapter 4 also deals with the rarity of events and shortness of tweets, this algorithm cannot capture their temporal and spatial variability due to the inability of DBSCAN (Density-based Spatial Clustering of Applications with Noise) to detect clusters in data of varying density. Therefore, WARBLE is not only an integrated solution to the problem of event detection, but it also address one of the major weaknesses of Tweet-SCAN.

The rest of the chapter is structured as follows. In Section 5.1, we introduce the WARBLE model in full detail. The learning scheme for the background model and the variational inference algorithm are described in Section 5.2. In Section 5.3, we learn WARBLE in “La Mercè 2014” data set and compare its detection performance against McInerney & Blei’s model (McInerney and Blei, 2014) and Tweet-SCAN. We conclude in Section 5.4 by presenting some remarks and future work.

## 5.1 WARBLE: the Probability Model

In this section we explain how the WARBLE model explicitly addresses rarity, variability and text-shortness. In the remaining,  $\mathbb{T}_n$  corresponds to a random variable which represents the time, geolocation and message of the  $n$ -th tweet, and  $\mathbb{T} = \{\mathbb{T}_1, \dots, \mathbb{T}_N\}$  is the full collection of observed tweets.

### 5.1.1 Addressing Rarity

The model proposed by [McInerney and Blei \(2014\)](#) is a mixture model in which every mixture component shares the same distributional form. Fig. 5.1a shows the PGM (Probabilistic Graphical Model) for their model where the  $N$  plate corresponds to the collection of tweets and the  $K$  plate, to the mixture components (i.e. events). Every mixture component is a probability distribution governed by a different global variable  $\beta_k$ . Besides, the mixture proportions  $\pi$  define another global variable on the  $K$ -simplex that assigns the proportion of tweets to each event. For each tweet  $\mathbb{T}_n$ , authors assume the existence of a latent event, encoded in the discrete hidden variable  $e_n$ , from which the data for the  $n$ -th tweet is generated as follows. Given  $e_n$ , the distribution of  $\mathbb{T}_n$  is

$$\mathbb{T}_n \sim f(\beta_{e_n}) \quad (5.1)$$

where  $f$  is the pdf, common for all mixture components. That is, the only difference between two events  $k$  and  $k'$  is that their parameters  $\beta_k$  and  $\beta_{k'}$  are different, but the functional form of  $f$  remains the same among components.

The joint probability distribution for McInerney and Blei's model can be expressed as follows,

$$p(\mathbb{T}, e, \beta, \pi) = p(\pi | \alpha_\pi) \prod_{n=1}^N p(\mathbb{T}_n | \beta_{e_n}) p(e_n | \pi) \prod_{k=1}^K p(\beta_k | \alpha_\beta) \quad (5.2)$$

where  $p(\pi | \alpha_\pi) = \text{Dir}(\alpha_\pi)$  follows a Dirichlet distribution,  $p(e_n | \pi) = \text{Cat}(\pi)$  is a Categorical distribution with parameters  $\pi$  and the functional form of  $p(\mathbb{T}_n | \beta_{e_n})$  is common for all  $K$  components. Moreover, the model considers a prior over the event parameters  $p(\beta_k | \alpha_\beta)$ .

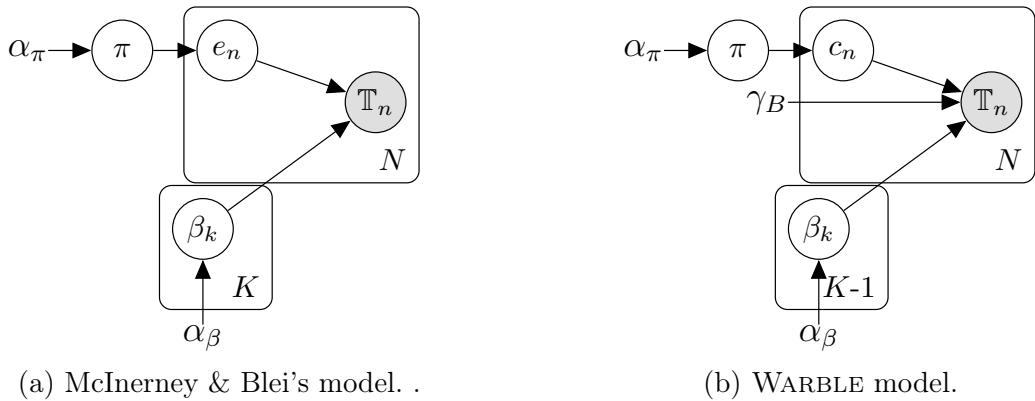


Figure 5.1: Simplified PGMs.

As discussed in the introduction, a vast majority of tweets is not event related. We would like to address rarity of event data by introducing a new mixture component, to which we

refer as *background*, which will group all those tweets which are not part of any event. In probabilistic terms, this means that the distribution of tweets inside the background component should be widely different from that inside events. McInerney and Blei’s model assumes (Eq. 5.1) that all components follow the same base distribution  $f$ , and thus it is unable to deal with the introduction of a background component whose distribution is widely different from that of events.

Accordingly, we propose to generalize McInerney and Blei’s model to handle heterogeneous components. To do that, for each component  $k$ , we enable a different base function  $f_k$  as shown in Eq (5.3).

$$\mathbb{T}_n \sim f_{e_n}(\beta_{e_n}). \quad (5.3)$$

Our model fits into the framework proposed by (Banfield and Raftery, 1993). To the best of our knowledge no application of that framework to event modelling has been reported.

The WARBLE model depicted in Fig. 5.1b is the PGM representation for an heterogeneous mixture model of tweets in which the  $K$ -th component (the background) follows a different statistical distribution. This component corresponds to the background and is represented through a set of parameters  $\gamma_B$ . Moreover, the latent assignments are now symbolized through  $c_n$  to denote that a tweet might be generated by event components ( $c_n < K$ ) or by background ( $c_n = K$ ).

The joint probability distribution for Fig. 5.1b can be written as,

$$p(\mathbb{T}, c, \beta, \pi) = p(\pi | \alpha_\pi) \prod_{n=1}^N p_{c_n}(\mathbb{T}_n | \beta_{c_n}, \gamma_B) p(c_n | \pi) \prod_{k=1}^{K-1} p(\beta_k | \alpha_\beta) \quad (5.4)$$

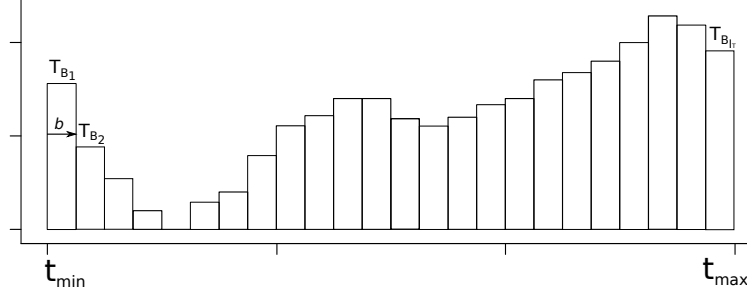
where now the tweet distribution depends on the component assignment,  $p_{c_n}(\mathbb{T}_n | \beta_{c_n}, \gamma_B)$ . Moreover, we observe that the background component does not consider a prior over its parameters. The next section provides additional details on how we model the distribution of the background component.

### 5.1.2 Addressing Variability

Geo-located social data such as tweets tends to be unevenly distributed through space and time. For example, it is known that users are more likely to tweet during late evening and from highly populated regions (Li et al., 2013). Because of this, we foresee the need to explicitly take this variability into account in order to identify events at peak hours as well as during valleys. The WARBLE model proposed in Fig. 5.1b enables to consider an arbitrary distribution with parameters  $\gamma_B$  for the background component. Here, we propose to model this background through two independent histogram distributions with parameters  $T_B$  and  $L_B$ , respectively.

The temporal histogram distribution can be represented through a piecewise-continuous function which takes constant values  $(T_{B_1}, T_{B_2}, \dots, T_{B_{I_T}})$  over the  $I_T$  contiguous intervals in the variable domain. For example, Fig. 5.2 shows the 1D-histogram distribution in the temporal range from  $t_{min}$  to  $t_{max}$ , in which there are  $I_T$  intervals of length  $b$ . Moreover, we must note that the piecewise function has to be normalised to sum 1 in order to fulfil the properties of probability distributions.

Similarly, the spatial background is modelled through a 2D-histogram distribution over the geographical space, which is represented in a Cartesian coordinate system. The 2d-

Figure 5.2: Temporal histogram distribution  $1d\text{-Hist}(\cdot)$ .

piecewise-continuous function is expressed through  $I_L$  constant values  $(L_{B_1}, L_{B_2}, \dots, L_{B_{I_L}})$  in a grid of squares with size  $b \times b$  each.

Through these histogram distributions, the WARBLE model can consider different spatio-temporal backgrounds which can be learned from tweets as we will see in Section 5.2.1.

### 5.1.3 Addressing Text-shortness

The shortness of tweets requires novel ways to address the lack of word co-occurrences to learn good topic representations. A common approach, seen in Section 4.1.3, consists in grouping tweets by hashtag or other related features (Hong and Davison, 2010). Others (McInerney and Blei, 2014) have leveraged on an external data set to perform transfer learning of topics. Here, we instead propose to address this issue in a probabilistic manner by clustering tweets into components  $c_n$  and learning component-specific topic proportions  $\theta_k$  at the same time. Integration of clustering and topic models has been studied for long-text (Xie and Xing, 2013) and short text (Quan et al., 2015) and shown that the integrated model improves both clustering and topic modelling.

$$\pi \sim \text{Dir}(\alpha_\pi)$$

For each component  $k = 1 \dots K$

$$\theta_k \sim \text{Dir}(\alpha_\theta)$$

For each topic  $t = 1 \dots T$

$$\phi_t \sim \text{Dir}(\alpha_\phi)$$

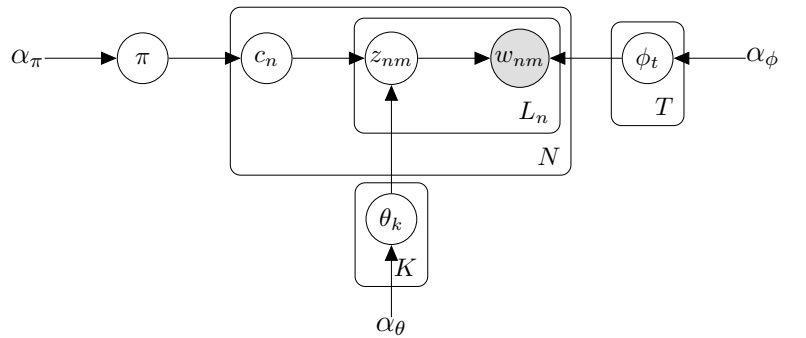
For each document  $n = 1 \dots N$

$$c_n \sim \text{Cat}(\alpha)$$

For each word  $m = 1 \dots L_n$

$$z_{nm} \sim \text{Cat}(\theta_{c_n})$$

$$w_{nm} \sim \text{Cat}(\phi_{z_{nm}})$$



Process 5.1: WARBLE TM.

Figure 5.3: WARBLE topic model (TM).

In Fig. 5.3, we show the PGM for the WARBLE topic model and Proc. 5.1 describes the generative process. To keep it simple, this graphical model only shows the variables related to the textual features of tweets. Hence, this model is a particular case of that

in Fig. 5.1b for the textual features of tweets. In fact, the textual content of the  $n$ -th tweet,  $\mathbb{T}_n$ , is represented here by the sequenced bag of words introduced in Section 2.3.1,  $\mathbf{w}_n = \{w_{n1}, \dots, w_{nM_n}\}$ . Moreover, the global variables  $\beta$  in Fig. 5.1b correspond here to the topic proportions,  $\theta = \{\theta_1, \dots, \theta_K\}$  and to the topic distributions,  $\phi = \{\phi_1, \dots, \phi_T\}$ . For both types of variables, the WARBLE considers prior distributions in the form of  $\text{Dir}(\phi_t|\alpha_\phi)$  and  $\text{Dir}(\theta_k|\alpha_\theta)$ , as defined by Eq. (B.8). Note also that the topic model does not consider a different background component for text, since the background is devoted to the spatio-temporal features whose variability can be better estimated.

The generative process for the local variables of the  $n$ -th tweet goes as follows. First, a component  $c_n$  is drawn from a Categorical distribution over the global component proportions  $\pi$ . For each word in the sequenced representation  $m = 1 \dots L_n$ , the topic assignment variable  $z_{nm}$  is then sampled from a Categorical distribution parametrised with the component-specific topic proportions  $\theta_{c_n}$ . Finally, each word  $w_{nm}$  is sampled from the corresponding topic distribution with parameter  $\phi_{z_{nm}}$ . Formally, the joint probability of the  $n$ -th tweet given all the global variables  $\pi$ ,  $\theta$  and  $\phi$  can be written as,

$$p(\mathbf{w}_n, \mathbf{z}_n, c_n | \pi, \theta, \phi) = \text{Cat}(c_n | \pi) \prod_{m=1}^{M_n} \text{Cat}(z_{nm} | \theta_{c_n}) \text{Cat}(w_{nm} | \phi_{z_{nm}}). \quad (5.5)$$

This topic model is different from traditional topic models like LDA (Latent Dirichlet Allocation), see PGM in Fig. 2.6, in the sense that the topic proportions  $\theta$  are not per-document, but per-component. Therefore, topic proportions in Fig. 5.3 are global variables that contain the topic proportions of the  $k$ -th cluster of tweets. Note that this distribution would be the same than that of a LDA model which has been trained with the documents pooled by these components. However, the main difference is that one has to jointly learn the clustering (i.e. assigning tweets to components  $c_n$ ) and the topic proportions per-component.

Next, we will show that by integrating this topic model into the complete WARBLE model, the assignments of component to tweets, i.e. clustering, can be done taking the spatio-temporal features and background into account. This might enable to obtain event-related topics from the event components, providing an interesting approach for automatic event summarisation (Long et al., 2011).

### 5.1.4 The Complete WARBLE Model

We present next the complete WARBLE model. The PGM in Fig. 5.4 provides a more detailed graph of the model depicted in Fig. 5.1b and also extends the topic model in Fig. 5.3 with the spatio-temporal part. Furthermore, we provide the complete generative process for Warble in Proc. 5.2.

In the complete WARBLE model tweets  $\mathbb{T}_n$  are represented by their temporal  $t_n$ , spatial  $l_n$  and textual  $\mathbf{w}_n$  features. The parameters  $\beta_k$  associated with the  $k$ -th event component comprise the set of variables  $\beta_k = \{\tau_k, \lambda_k, \mu_k, \Delta_k, \theta_k\}$ . As for the hyperparameters,  $\alpha_\beta$  in Fig. 5.1b, now corresponds to the set of hyperparameters  $m_\tau, \beta_\tau, a_\lambda, b_\lambda, m_\mu, \beta_\mu, \nu_\Delta, W_\Delta, \alpha_\theta$  in Fig. 5.4. Finally, the hyperparameter of the background component  $\gamma_B$  in Fig. 5.1b is composed of the hyperparameters for the temporal ( $T_B$ ) and spatial ( $L_B$ ) features in Fig. 5.4. Note that the complete WARBLE model also contains the global topic variables,  $\phi$ , and their hyperparameters,  $\alpha_\phi$ .

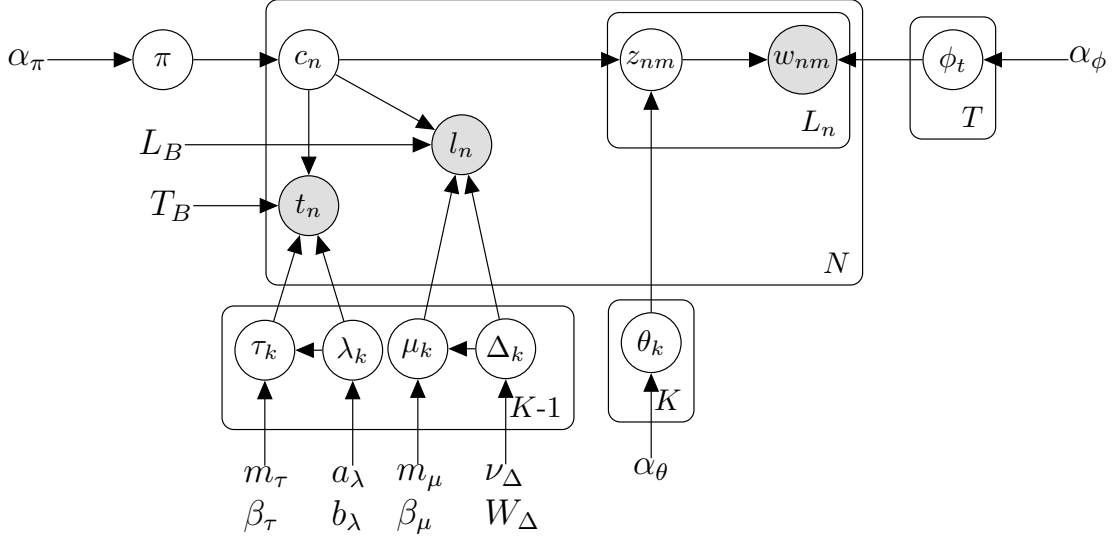


Figure 5.4: The WARBLE model in detail.

In Eq. (5.6) we provide the joint probability distribution, which fully describes the WARBLE model in probabilistic terms.

$$p(\mathbb{T}, c, \beta, \pi, \phi, \theta_K) = p(\pi|\alpha_\pi)p(\phi|\alpha_\phi) \prod_{n=1}^N p_{c_n}(\mathbb{T}_n | \beta_{c_n}, \gamma_B) p(c_n|\pi) \prod_{k=1}^{K-1} p(\beta_k|\alpha_\beta) p(\theta_K|\alpha_\beta) \quad (5.6)$$

In the remaining we specify each of the factors in the right hand side of Eq. (5.6). As discussed earlier,  $p(\pi|\alpha_\pi)$  follows a Dirichlet distribution, that is  $p(\pi|\alpha_\pi) = \text{Dir}(\pi|\alpha_\pi)$  and  $p(\phi|\alpha_\phi) = \prod_{t=1}^T \text{Dir}(\phi_t|\alpha_\phi)$  is the product of  $T$  Dirichlet distributions with hyperparameter  $\alpha_\phi$ .

As for the tweet probability distribution  $p_{c_n}(\mathbb{T}_n | \beta_{c_n}, \gamma_B)$ , we have that

$$p_{c_n}(\mathbb{T}_n | \beta_{c_n}, \gamma_B) = p_{c_n}(t_n | \tau_{c_n}, \lambda_{c_n}, T_B) p_{c_n}(l_n | \mu_{c_n}, \Delta_{c_n}, L_B) p(\mathbf{w}_n | \theta_{c_n}, \phi) \quad (5.7)$$

Here, the posting time  $t_n$  of event-related tweets arises from a Normal distribution  $N(\cdot)$  with unknown mean  $\tau_{c_n}$  and precision  $\lambda_{c_n}$ , and that of non-event tweets is generated by a 1D histogram distribution  $\text{Hist}(\cdot)$  with parameter  $T_B$ , formally

$$p_{c_n}(t_n | \tau_{c_n}, \lambda_{c_n}, T_B) = \begin{cases} \text{Hist}(t_n | T_B), & \text{if } c_n = K \\ N(t_n | \tau_{c_n}, \lambda_{c_n}), & \text{otherwise.} \end{cases} \quad (5.8)$$

Similarly, the geographical locations  $l_n$  of event-related tweets comes from a multivariate Normal distribution with unknown mean  $\mu_{c_n}$  and precision  $\Delta_{c_n}$  and that of non-event tweets is generated by a 2D histogram distribution  $\text{Hist}(\cdot)$  with parameter  $L_B$ :

$$p_{c_n}(l_n | \mu_{c_n}, \Delta_{c_n}, L_B) = \begin{cases} \text{Hist}(l_n | L_B), & \text{if } c_n = K \\ N(l_n | \mu_{c_n}, \Delta_{c_n}), & \text{otherwise.} \end{cases} \quad (5.9)$$

Finally, the bag of words  $\mathbf{w}_n = \{w_{n1}, \dots, w_{nM_n}\}$  for event and non-event components are generated according to the topic model described by Eq. (5.5). That is,

$$p(\mathbf{w}_n | \theta_{c_n}, \phi) = \prod_{m=1}^{L_n} \sum_{z_{nm}} \text{Cat}(z_{nm} | \theta_{c_n}) \text{Cat}(w_{nm} | \phi_{z_{nm}}). \quad (5.10)$$

The prior over event component parameters  $p(\beta_k | \alpha_\beta)$  is

$$p(\beta_k | \alpha_\beta) = N(\mu_k | m_\mu, \beta_\mu \Delta_k) W(\Delta_k | \nu_\Delta, W_\Delta) N(\tau_k | m_\tau, \beta_\tau \lambda_k) \text{Ga}(\lambda_k | a_\lambda, b_\lambda) \text{Dir}(\theta_k | \alpha_\theta) \quad (5.11)$$

where the unknown means and precisions are drawn from a Normal-Gamma  $N(\cdot)\text{-Ga}(\cdot)$  and a Normal-Wishart  $N(\cdot)\text{-W}(\cdot)$ , which are conjugate priors to the uni-variate and multivariate Normal, respectively. As explained in Section 5.1.3, the topic proportions of the background component also follow the same Dirichlet than the event-related components,  $\text{Dir}(\theta_k | \alpha_\phi)$ , which are also conjugate to the Categorical.

|   |   |
|---|---|
| $\pi \sim \text{Dir}(\alpha_\pi)$<br>$\theta_K \sim \text{Dir}(\alpha_\theta)$<br>For each event component $k = 1 \dots K - 1$<br>$\mu_k \sim N(m_\mu, \beta_\mu \Delta_k)$<br>$\Delta_k \sim W(\nu_\Delta, W_\Delta)$<br>$\tau_k \sim N(m_\tau, \beta_\tau \lambda_k)$<br>$\lambda_k \sim \text{Ga}(a_\lambda, b_\lambda)$<br>$\theta_k \sim \text{Dir}(\alpha_\theta)$<br>For each topic $t = 1 \dots T$<br>$\phi_t \sim \text{Dir}(\alpha_\phi)$ | For each tweet $n = 1 \dots N$<br>$l_n \sim p_{c_n}(l_n   \mu_{c_n}, \Delta_{c_n}, L_B)$<br>$t_n \sim p_{c_n}(t_n   \tau_{c_n}, \lambda_{c_n}, T_B)$<br>$c_n \sim \text{Cat}(\alpha)$<br>For each word $m = 1 \dots L_n$<br>$z_{nm} \sim \text{Cat}(\theta_{c_n})$<br>$w_{nm} \sim \text{Cat}(\phi_{z_{nm}})$ |
|---|---|

Process 5.2: The WARBLE generative process in detail.

## 5.2 Learning Scheme

In this section we describe how to use the WARBLE model to identify a set of events in a region during a period of interest. The procedure assumes the availability of a recorded data set of tweets from that region and follows two steps. First, we use the tweets previous to the start of the period of interest to derive a background model. Then, we use the tweets recorded during the period of interest to find the most probable assignment of tweets to mixture components.

### 5.2.1 Learning the Background Component

To learn the spatio-temporal background from tweets, we propose to collect tweets previous to the period of interest and within the same region in order to add a sense of typicality to the model.

From the collected tweets, the temporal background is built by first computing the daily histogram with  $I_T$  bins. Then, the daily histogram is smoothed by means of a low pass



filter that removes the high frequency components. The filter is constructed by multiplying the signal in the Fourier or frequency domain by a rectangular function with a certain cut-off frequency  $f_c$ . The smoothed histogram is finally normalised and its parameters  $T_{B_1}, T_{B_2}, \dots, T_{B_{I_T}}$  correspond to the temporal background.

The spatial background is build following the same procedure. However, geographical location has to be first projected into a Cartesian coordinate system in order to consider locations in a 2D Euclidean space. The spatial range limits can be determined from the most southwestern and northeastern points. We consider instead a two dimensional Gaussian filter with a standard deviation  $\sigma$ . The smoothed 2D-histogram provides the parameters for the spatial background  $L_{B_1}, L_{B_2}, \dots, L_{B_{I_L}}$ .

Therefore, the number of bins for the temporal  $I_T$  and spatial  $I_L$  histograms as well as the cut-off frequency  $f_c$  and standard deviation  $\sigma$  for the low pass filters are hyperparameters that will be set during the experimentation.

### 5.2.2 Assigning Tweets to Mixture Components

We are interested in finding the most probable assignment of tweets to mixture components, given the data at hand, that is finding  $\mathbf{c}^*$

$$\mathbf{c}^* = \underset{\mathbf{c}}{\operatorname{argmax}} p(\mathbf{c} | \mathbf{l}, \mathbf{t}, \mathbf{w}; \Gamma) \quad (5.12)$$

where  $\Gamma$  stands for the model hyperparameters  $L_B, T_B, \alpha_\pi, \alpha_\theta, \alpha_\phi, m_\tau, \beta_\tau, a_\lambda, b_\lambda, m_\mu, \beta_\mu, \nu_\Delta$  and  $W_\Delta$ . Exactly assessing  $\mathbf{c}^*$  is computationally intractable for the WARBLE model. Therefore, we propose to

1. Use mean-field variational Bayesian inference, presented in Section 2.5.2, to approximate  $p(X|D; \Gamma)$  (where  $X$  stands for the set of random variables containing  $\mathbf{c}, \mathbf{z}, \pi, \tau, \lambda, \mu, \Delta, \theta$  and  $\phi$ , and  $D$  stands for our data, namely  $\mathbf{l}, \mathbf{t}$ , and  $\mathbf{w}$ ) by a distribution  $q(X; \eta)$  (where  $\eta$  stands for the variational parameters to be detailed later).
2. Assess  $\mathbf{c}^*$  from the approximation, that is

$$\mathbf{c}^* = \underset{\mathbf{c}}{\operatorname{argmax}} q(\mathbf{c}; \eta) = \underset{\mathbf{c}}{\operatorname{argmax}} \int_{X=\mathbf{c}} q(X; \eta). \quad (5.13)$$

In the following we provide detail on each of these two points.

#### 5.2.2.1 Mean-Field Variational Bayesian inference

Our mean-field variational inference algorithm relies on minimising the KL (Kullback-Leibler) divergence between  $p(X|D; \Gamma)$  and a distribution  $q(X; \eta)$  which factorises as

$$\begin{aligned} q(X; \eta) &= q(\pi) \prod_{t=1}^T q(\phi_t) \prod_{n=1}^N q(c_n) \prod_{m=1}^{M_n} q(z_{nm}) \\ &\quad q(\theta_K) \prod_{k=1}^{K-1} q(\tau_k) q(\lambda_k) q(\mu_k) q(\Delta_k) q(\theta_k). \end{aligned} \quad (5.14)$$

The KL divergence is minimised through an iterative coordinate-descent scheme until convergence is reached. Thus, the factors in Eq. (5.14) are sequentially updated, one factor at a time. The mean-field variational update for the factor corresponding to a random variable  $x$  whatsoever is

$$q(x) \propto \exp \left( \int_{X-x} q(X; \eta) \log p(X, D; \Gamma) \right) \quad (5.15)$$

where  $\log p(X, D; \Gamma)$  is the logarithm of the join probability distribution for the WARBLE model defined in Eq. (5.6). After all variables have been updated the KL divergence is compared with that of the previous iteration. In case convergence has not been reached yet, another round of updates is started.

We notice that due to the introduction of the background distributions, the model is not fully conditionally conjugate as the LVM (Latent Variable Model) discussed in Section 2.5.2. Thus, the updates in Eq. (5.15) need to be manually derived for each variable. To exemplify the derivations, we include here the development of the most complex update, that of the assignment variable  $c_n$ . Since our distribution follows the Bayesian network in Fig. 5.4, Eq. (5.15) can be simplified to

$$q(c_n) \propto \exp \left( \int_Z q(Z) \log p(c_n, Z, D; \Gamma) \right) \quad (5.16)$$

where  $Z$  is the set of variables in the Markov blanket of  $c_n$ , which are  $\pi$ ,  $t_n$ ,  $\tau$ ,  $\lambda$ ,  $l_n$ ,  $\mu$ ,  $\Delta$ ,  $z_{n..}$  and  $\theta$ .

Given that the right side of Eq. (5.16) is proportional to the approximate distribution  $q(c_n)$ , we can disregard terms that do not depend on  $c_n$  and express the remaining as a product,

$$q(c_n) \propto f_{\text{prior}}(c_n) \cdot f_{\text{time}}(c_n) \cdot f_{\text{loc}}(c_n) \cdot \prod_{m=1}^{M_n} f_{m\text{-word}}(c_n) \quad (5.17)$$

where

$$\begin{aligned} f_{\text{prior}}(c_n) &= \exp \left( \int_{\pi} q(\pi) \log p(c_n | \pi) \right) \\ f_{\text{time}}(c_n) &= \exp \left( \int_{\tau_{c_n}, \lambda_{c_n}} q(\tau_{c_n}) q(\lambda_{c_n}) \log p(t_n | \tau_{c_n}, \lambda_{c_n}) \right) \\ f_{\text{loc}}(c_n) &= \exp \left( \int_{\mu_{c_n}, \Delta_{c_n}} q(\mu_{c_n}) q(\Delta_{c_n}) \log p(l_n | \mu_{c_n}, \Delta_{c_n}) \right) \\ f_{m\text{-word}}(c_n) &= \exp \left( \int_{\theta_{c_n}, z_{nm}} q(\theta_{c_n}) q(z_{nm}) \log p(z_{nm} | \theta_{c_n}) \right). \end{aligned} \quad (5.18)$$

We observe that there are four factors, one for the mixture proportions and one for each tweet feature (posting time, geographical location and text message).

Since  $c_n$  is a discrete variable,  $q(c_n)$  fits in the functional form of a Categorical distribution with variational parameter  $c'_{nk}$ , defined as the normalisation of  $\tilde{c}'_{nk}$ ,

$$c'_{nk} = \frac{\tilde{c}'_{nk}}{\sum_{k=1}^K \tilde{c}'_{nk}} \quad (5.19)$$

| $q(x)$         | Functional form  |
|----------------|--|
| $q(\pi)$       | $\text{Dir}(\pi \pi'_k)$   |
| $q(c_n)$       | $\text{Cat}(c_n c'_{nk})$  |
| $q(z_{nm})$    | $\text{Cat}(z_{nm} z'_{nmt})$  |
| $q(\phi_t)$    | $\text{Dir}(\phi_t \phi'_t)$   |
| $q(\tau_k)$    | $\text{N}(\tau_k m_{\tau_k}, \beta'_{\tau_k} \frac{a'_\lambda}{b'_\lambda})$ |
| $q(\lambda_k)$ | $\text{Ga}(\lambda_k a'_\lambda, b'_\lambda)$                                |
| $q(\mu_k)$     | $\text{N}(\mu_k \mu'_k, \beta'_{\mu_k} \nu' W')$                             |
| $q(\Delta_k)$  | $\text{W}(\Delta_k \nu', W')$  |
| $q(\theta_k)$  | $\text{Dir}(\theta_k \theta'_k)$   |

Table 5.1: Functional forms for  $q(X)$ .

where  $\tilde{c}'_{nk}$  can be obtained from Eq. (5.17):

$$\tilde{c}'_{nk} = f_{\text{prior}}(k) \cdot f_{\text{time}}(k) \cdot f_{\text{loc}}(k) \cdot \prod_{m=1}^{M_n} f_{m\text{-word}}(k). \quad (5.20)$$

Note that the background component takes no part in  $f_{\text{prior}}$  and  $f_{m\text{-word}}$ , whose expressions can hence be derived following a standard procedure described in Section 2.5.2. Thus, we omitted them next.

However, the introduction of a background model entails differences in the spatio-temporal factors  $f_{\text{loc}}$  and  $f_{\text{time}}$ , since the background component ( $k = K$ ) follows a different distribution function. Considering the pdf in Eq. (5.8), the temporal factor can be defined as follows,

$$f_{\text{time}}(k) = \begin{cases} \text{Hist}(t_n|T_B), & k = K \\ \exp\left(\int_{\tau_k, \lambda_k} q(\tau_k)q(\lambda_k) \log \text{N}(t_n|\tau_k, \lambda_k)\right), & \text{otherwise} \end{cases} \quad (5.21)$$

and from Eq. (5.9), the spatial factor is,

$$f_{\text{loc}}(k) = \begin{cases} \text{Hist}(l_n|L_B), & k = K \\ \exp\left(\int_{\mu_k, \Delta_k} q(\mu_k)q(\Delta_k) \log \text{N}(l_n|\mu_k, \Delta_k)\right), & \text{otherwise} \end{cases} \quad (5.22)$$

where in each equation the event components are computed from the corresponding Normal distributions and the background component from the Histogram distribution.

Nonetheless, to find a closed-form expression for Eq. (5.21) we need to derive the approximated distributions for  $q(\tau_k)$  and  $q(\lambda_k)$ . We provide a summary of the functional forms for each variational distribution  $q(x)$  in Table 5.1. Full details on the updates can be found in Appendix C.

### 5.2.2.2 Assigning Tweets to Components through Variational inference

Recall that our objective was to find the most likely assignment of tweets to mixture components using the variational approximation to the posterior shown in Eq. (5.13). Note that

we can take benefit from the fact that  $q(X)$  factorises as shown in Eq. (5.14) to assess the mixture component for each tweet independently. Thus, the  $n$ -th tweet will be assigned to the mixture component which maximises the Categorical distribution  $q(c_n; c'_n)$ , that is,

$$c_n^* = \operatorname{argmax}_{c_n} q(c_n; c'_n) = \operatorname{argmax}_k c'_{nk}. \quad (5.23)$$

This means that tweets will be assigned to the most likely component according to the probabilities given by Eq. (5.19). If the  $K$ -th component of  $c'_n$  is the largest, then the  $n$ -th tweet is assigned to the background; otherwise, it is assigned to one of the  $K - 1$  events.

## 5.3 Experimentation

In this section, we present how to setup WARBLE for detecting events in “La Mercè 2014” data set introduced earlier in Section 3.3. We then evaluate the detection performance of WARBLE and compare to other event detection methods in terms of extrinsic clustering metrics introduced in Section 2.6.2. The code to reproduce all the experiments can be found in this repository<sup>1</sup>.

### 5.3.1 WARBLE Settings for “La Mercè 2014”

In this section, we detail the parameters of the WARBLE model as well as the spatio-temporal backgrounds for “La Mercè 2014”. We focus on the 2014 edition because it contains more tweets with exact geo-location than “La Mercè 2015”. Besides we restrict this study to a particular day, the 24th of September 2014, when most of the labelled events occurred. Therefore, the data set is composed of 2173 tweets out of which 202 belong to 6 distinct real-world events in Tab 3.2. These are the music concert at Bogatell beach area, the human towers exhibition at Plaça Sant Jaume, the open day at MACBA museum, the food market at Parc de la Ciutadella, the wine tasting fair at Arc de Triomf and the fireworks near Plaça d’Espanya. Moreover, we have also identified a 7th anomalous increase of tweets in the Bogatell area during the afternoon as a result of several users reviving the earlier concert.

The WARBLE model presented in Section 5.1 contains several hyperparameters. Although their optimization is left for future work, we have not experimented substantial differences in the results when changing them. The number of components  $K$  is set to 8 so that the model is able to capture the 7 previously identified events occurring during the 24th of September 2014. Following the results from the previous chapter with the non-parametric topic model, we set the number of topics  $T$  to 30. Table 5.2 shows the values used for WARBLE in “La Mercè 2014” data set and for the rest of probabilistic methods that we compare later.

---

<sup>1</sup><https://github.com/jcapde/WARBLE>

| $K$ | $T$ | $\alpha_\pi$ | $\alpha_\theta$ | $\alpha_\phi$ | $m_\tau$    | $\beta_\tau$                        | $a_\lambda$ | $b_\lambda$ | $m_\mu$     | $\beta_\mu$                         | $\nu_\Delta$ | $W_\Delta$       |
|-----|-----|--------------|-----------------|---------------|-------------|-------------------------------------|-------------|-------------|-------------|-------------------------------------|--------------|------------------|
| 8   | 30  | 0.1          | 0.1             | 0.1           | $\bar{t}_n$ | $\frac{9}{10\ t_{max}-t_{min}\ ^2}$ | 100         | 1           | $\bar{l}_n$ | $\frac{9}{10\ l_{max}-l_{min}\ ^2}$ | 100          | $I_{2 \times 2}$ |

Table 5.2: Hyperparameter settings.  $\bar{t}_n$  and  $\bar{l}_n$  correspond to the average values across temporal and spatial features.  $t_{max}$ ,  $t_{min}$  and  $l_{max}$ ,  $l_{min}$  to the maximum and minimum values in the corresponding dimensions.  $I_{2 \times 2}$  is a  $2 \times 2$  identity matrix.

In addition to the tweets of the 24th of September, we also consider tweets previous to the period of interest in order to learn the spatio-temporal backgrounds  $T_B$  and  $L_B$  as explained in Section 5.2.1. In particular, we used tweets from the 20th to the 23th of September 2014 to build the spatio-temporal distributions as described next.

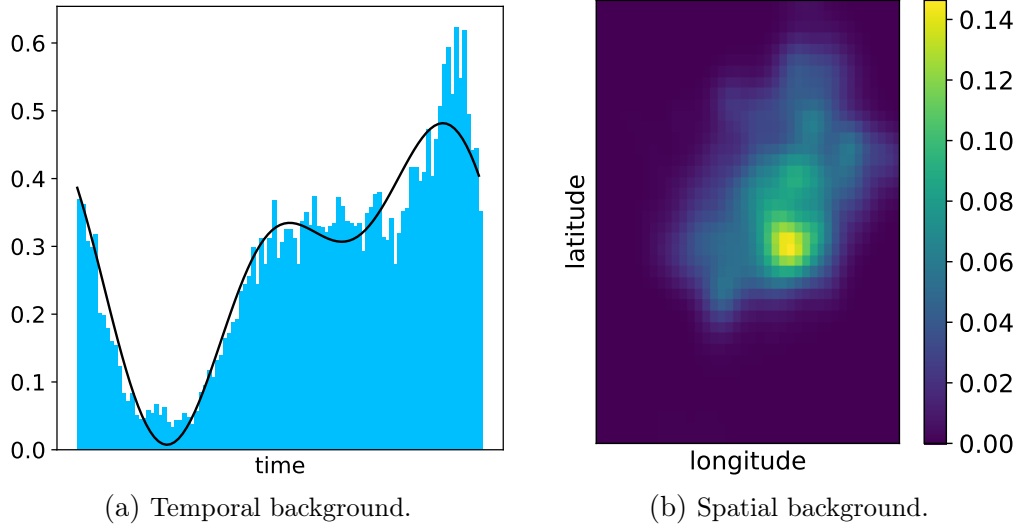


Figure 5.5: Spatio-temporal backgrounds.

Fig. 5.5a shows the daily histogram of tweets in which we observe a valley during the early morning and a peak at night, indicating low and high tweeting activity during these hours, respectively. The 1D histogram has been computed with  $I_T = 100$  bins and a cut-off frequency  $f_c = 0.6$ . Fig. 5.5a also contains the smoothed histogram distribution (black line) that is used to set the temporal background parameters  $T_{B_1}, T_{B_2}, \dots, T_{B_{I_T}}$ .

Fig. 5.5b is the smoothed histogram for all tweet locations, which give us the parameters for the spatial background  $L_{B_1}, L_{B_2}, \dots, L_{B_{I_L}}$ . The 2D histogram has been computed with  $I_L = 1600$  bins and a standard deviation of  $\sigma = 1.5$ . We observe that the most likely areas in the filtered histogram (in bright yellow) correspond to highly dense regions of Barcelona like the city center, while city surroundings are coloured in blue indicating lower density of tweets.

We note that the above backgrounds are in accordance with spatio-temporal behaviours founds in other studies (Li et al., 2013) and we did not experience significant changes when varying the hyperparameters specified above.

### 5.3.2 Results

First, we assess WARBLE in “La Mercè 2014” data set through recall figures for each labelled event. Then, we compare its performance against related methods such as McInerney & Blei model (McInerney and Blei, 2014) and Tweet-SCAN.

#### 5.3.2.1 Assessment of WARBLE in “La Mercè 2014”

Table 5.3 summarises the events that WARBLE was able to discover during the 24th of September in “La Mercè 2014” data set.

For each event, the set matching recall provides the fraction of relevant tweets that are correctly identified and BCubed recall, shown in parentheses, provides the average correctness. Despite their differences, both recall figures show very similar results. We observe that larger events (# tweets), such as concert and fireworks, are correctly identified (high recall) while smaller ones, like museums open day or human towers exhibition, are harder to detect. However, we notice that the food market and wine tasting exposition could not be discovered at all. We argue that this is because both were all-day events and had fewer tweets in comparison to the rest. Future work could explore to treat all-day events differently, for instance introducing priors for these events with greater temporal variance. Finally, the resulting mean coordinates (lat, long) and times from the probabilistic model are also coherent with “La Mercè” schedule.

| Event            | Proportion<br>of tweets | Recall<br>(BCubed) | Time<br>(hh:mm:ss)     | Location<br>(lat;long)                       |
|------------------|-------------------------|--------------------|------------------------|--|
| Concert          | 27/28                   | 0.96 (0.93)        | 02:32:40 $\pm$ 0:11:32 | 41.3931 $\pm$ 0.0014;<br>2.2058 $\pm$ 0.0018 |
| Human towers     | 11/20                   | 0.55 (0.36)        | 12:46:56 $\pm$ 0:08:40 | 41.3834 $\pm$ 0.0013;<br>2.1775 $\pm$ 0.0016 |
| Concert revival  | 26/30                   | 0.86 (0.76)        | 13:44:19 $\pm$ 0:10:17 | 41.3926 $\pm$ 0.0012;<br>2.2056 $\pm$ 0.0017 |
| Museums open day | 18/25                   | 0.72 (0.56)        | 18:18:33 $\pm$ 0:08:27 | 41.3836 $\pm$ 0.0012;<br>2.1716 $\pm$ 0.0044 |
| Fireworks        | 62/65                   | 0.95 (0.91)        | 22:11:10 $\pm$ 0:06:18 | 41.3734 $\pm$ 0.0015;<br>2.1496 $\pm$ 0.0022 |

Table 5.3: Recall figures and spatio-temporal features per event.

The WARBLE model, apart from spatio-temporal information, also provides information about which topics are linked to each event, as per the topic model presented in Section 5.1.3. Topic distributions plotted in Fig. 5.6, show that each event is mainly about one topic, except for the last one which corresponds to background component ( $k = K$ ) and its a mix of lots of topics. Therefore, there are two events whose main topic is number 17, one event for topic 24, another for topic 5 and one last event which is mainly about topic 14.

The content of each topic can be found in the corresponding topic distribution. Table 5.4 shows the most probable words for each topic, enabling us to understand the relationship between topics and events. For example, Topic 17 refers to music since words *concert*, *txarango* (local band) and *manel* (local band) are very likely. We have already seen that this topic was linked to two resulting events in Fig. 5.6 which we can associated with the music concert at *Bogatell* beach area and the revival on the afternoon. We also note that

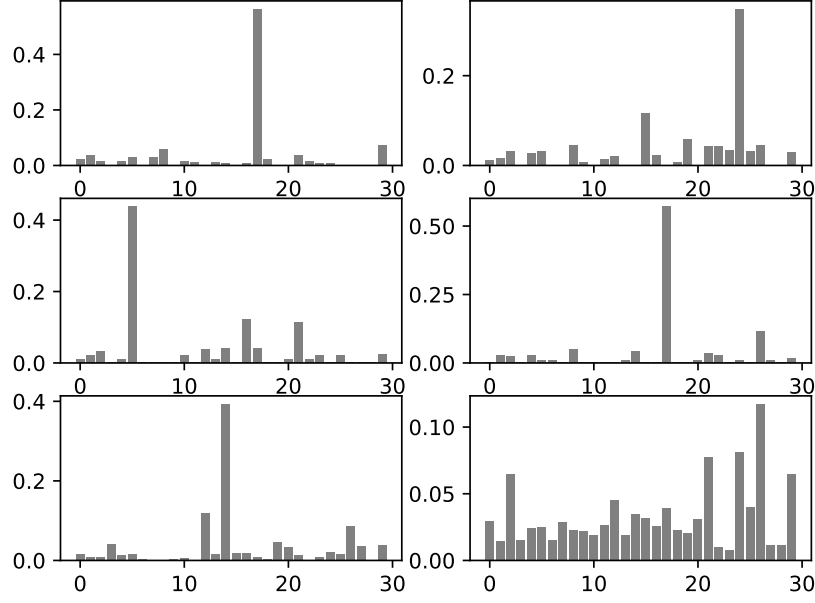


Figure 5.6: Topic proportions ( $\theta'_{k:}/\sum_t \theta'_{kt}$ ) per component (5 detected events and 1 background).

top words in each topic usually refer to the event location, which can be explained from the fact that most tweet messages explicitly mention the place. A complete list of the 10 most probable words for each topic can be found in Appendix D.1.

Table 5.4: 5 most probable words per topic  $\phi'_{t:}/\sum_v \phi'_{tv}$ .  
English translations in *italics*.

| Topic 5             | Topic 14          | Topic 17        | Topic 24         | Topic 26     |
|---------------------|-------------------|-----------------|------------------|--------------|
| museu               | piromusical       | platja          | plaça            | im           |
| <i>museum</i>       | <i>fireworks</i>  | <i>beach</i>    | <i>square</i>    | <i>I'm</i>   |
| macba               | plaça             | bogatell        | dia              | q            |
| <i>MACBA</i>        | <i>square</i>     | <i>Bogatell</i> | <i>day</i>       | <i>that</i>  |
| contemporani        | despanya          | txarango        | jaume            | gran         |
| <i>contemporary</i> | <i>from Spain</i> | <i>Txarango</i> | <i>Jaume</i>     | <i>big</i>   |
| fan                 | font              | concert         | catalunya        | mercé        |
| <i>do</i>           | <i>fountain</i>   | <i>concert</i>  | <i>Catalonia</i> | <i>Mercé</i> |
| veient              | poder             | manel           | day              | hoy          |
| <i>looking</i>      | <i>power</i>      | <i>Manel</i>    | <i>day</i>       | <i>today</i> |

By simultaneous learning of topics and events, we observed that event-related components contain event-specific topics which are very different from those in the background component. As we show next, this feature improves the discrimination capabilities of WARBLE.

### 5.3.2.2 Evaluation against State-of-the-art

In what follows, we compare WARBLE from Section 5.1 against other event detection techniques. In particular, we will compare the performance of:

- (A) McInerney & Blei model, which does not consider background and does not perform simultaneous topic-event learning.
- (B) The WARBLE model without simultaneous topic-event learning.
- (C) The WARBLE model without modelling background.
- (D) The complete WARBLE model.
- (E) Tweet-SCAN with  $\epsilon_1 = 250m$ ,  $\epsilon_2 = 3600s$ ,  $\epsilon_3 = 0.9$ ,  $\mu = 0.5$  as the best performing configuration in Chapter 4, and  $MinPts = 7$ , according to Fig. 5.7.

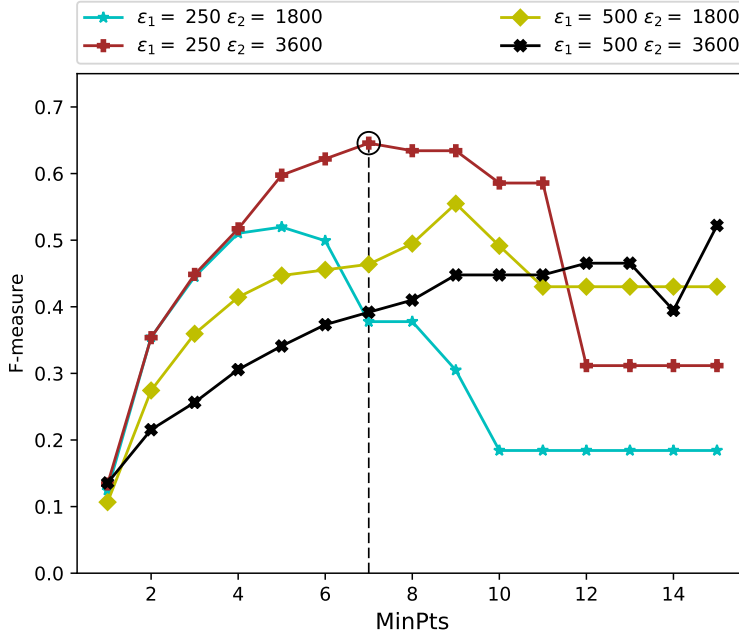


Figure 5.7: Best-performing Tweet-SCAN configuration on the 24th of September in “La Mercè” 2014.

For those models that do not perform simultaneous topic-event learning, the LDA model (Blei et al., 2003) is separately trained with tweets aggregated by key terms as proposed in (Hong and Davison, 2010). That is the case for models (A), (B) and (E).

Fig. 5.8a shows the results for each event detection model introduced earlier in terms of set matching metrics, whereas Fig. 5.8b shows the same experiments with the BCubed metrics. Results show that WARBLE outperforms the existing state-of-the-art models (A & E) in terms of F-measure and purity. Moreover, by analyzing the results of models B and C we see a clear synergy between background modelling and simultaneous topic-event learning. Neither of them separately achieves a large increase of the F-measure, but when combined they do. The same conclusions can be drawn from the analysis of BCubed metrics.

Fig. 5.9 provides visual insight on the quality of the events detected by each of the alternatives, by drawing tweets in a 3-dimensional space corresponding to the spatial (lat, long) and temporal (time) features. Each tweet is colored with the maximum likelihood event assignment ( $c_n^*$ ) for that tweet. Moreover, to improve visualization, the most populated cluster, which usually is the background, is plotted with tiny dots for all models,



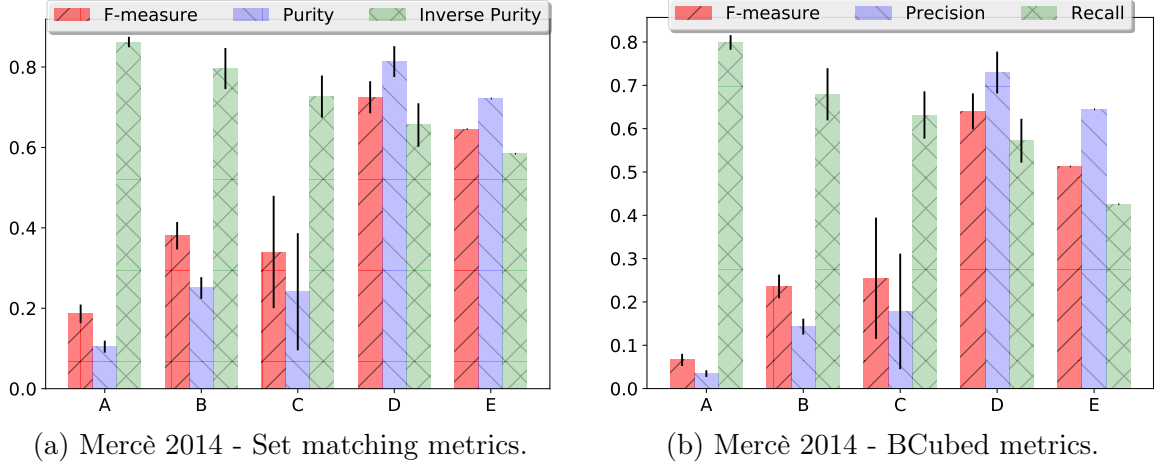


Figure 5.8: F-measure detection performance. (A) McInerney & Blei model (B) WARBLE w/o simultaneous topic-event learning (C) WARBLE w/o background model (D) WARBLE model (E) Tweet-SCAN.

except model A, which fails to capture a clear background cluster. The figure shows that the similarity between hand-labelled data and the WARBLE model can only be compared to that of Tweet-SCAN.

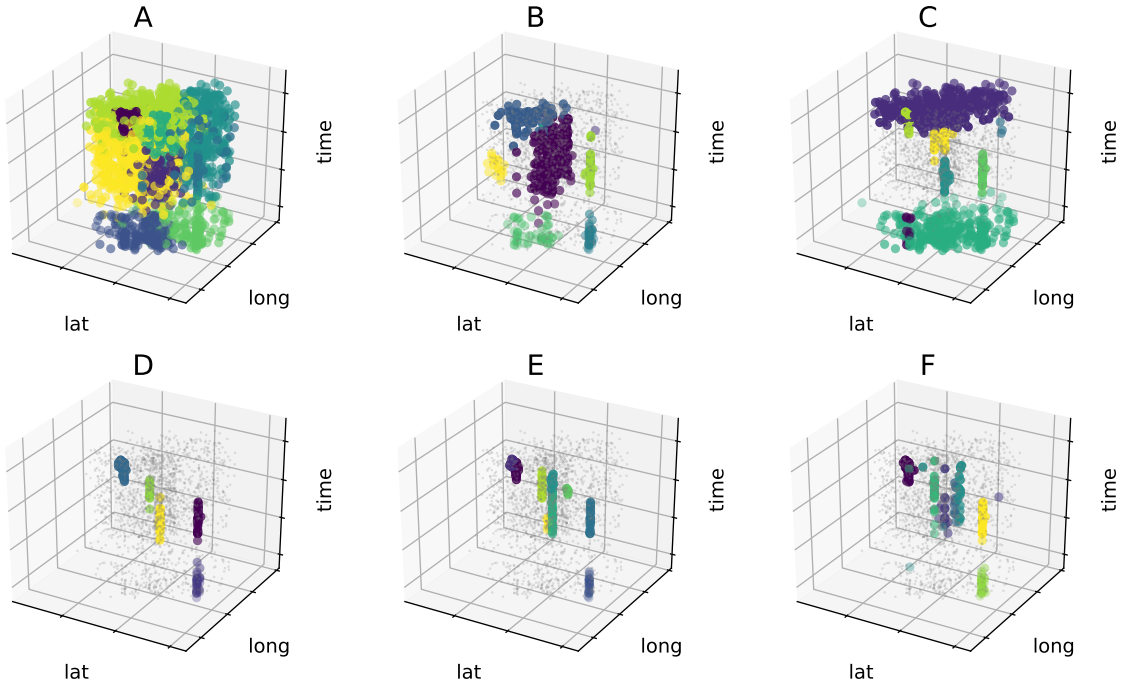


Figure 5.9: Visual comparison of results. (A) McInerney & Blei model (B) WARBLE w/o simultaneous topic-event learning (C) WARBLE w/o background model (D) WARBLE model (E) Tweet-SCAN (F) Labelled events

## 5.4 Summary and Conclusion

In this chapter, we identified three main challenges in event detection from Twitter data, namely rarity, text-shortness and variability. In order to address them, we proposed WARBLE, a new probabilistic model and variational learning algorithm that uncovers real-world events from tweets in an unsupervised manner. The WARBLE model explicitly tackles rarity and variability through a background component, which captures varying tweet densities in time and space. To mitigate text-shortness, our proposal simultaneously learn topics and events making it easier to find word co-occurrences among tweets that belong to the same event. Furthermore, this probabilistic approach to event detection paves the way to reason about unseen observations or partially observed data in a probabilistically well-principled way.

The experimental results show that WARBLE outperforms other techniques in detecting local events from “La Mercè 2014” data set. In particular, we observe that WARBLE outperforms the McInerney & Blei’s model thanks to the use of spatio-temporal backgrounds and the simultaneous learning of topics and events. As shown in the results, the performance of WARBLE is also slightly superior to that of Tweet-SCAN due to the extra capacity to deal with varying tweet densities in space and time and the simultaneous learning of topics and events. WARBLE allows users to define a spatio-temporal background that can handle changes in these dimensions (i.e. people tweeting more at night than at midday or at the city center, than at residential areas). Besides, the simultaneous learning of topics and events enables to better specify both components and increase the overall precision. Despite WARBLE contains 13 hyperparameters, Table 5.2, and Tweet-SCAN only 5, some of WARBLE’s hyperparameters are less sensitive to changes because they are parameters of prior distributions which become less important with increasing amounts of data. Nonetheless, Tweet-SCAN uncovers the number of events  $K$  from data and parametrisation, while this hyperparameter has to be set beforehand in WARBLE. Finally, we showed that the proposed model also provides automatic summarisation about events, enabling to describe different aspects of events, such as when and where it took place and what was about.

# Part II

## Likelihood Evaluation





# Likelihood Estimation in Poisson Factor Analysis

*“Without proper self-evaluation, failure is inevitable”*

John WOODEN

Capdevila, J., Cerquides, J., Torres, J., Petitjean, F., and Buntine, W. (2018c). A left-to-right algorithm for likelihood estimation in gamma-poisson factor analysis. In *Joint European Conference on Machine Learning and Knowledge Discovery in Databases*, pages 638–654. Springer

As shown in the previous chapters, topic models can be built in bigger models or methods to perform a particular task, and hence their performance can be evaluated extrinsically, i.e. in terms of the task. However, when comparing across different topic models, one usually prefers to evaluate their performance intrinsically, or in other words, independently of the task at hand. The intrinsic evaluation of topic models has attracted the interest of the community and several estimation methods have been developed for the popular LDA (Latent Dirichlet Allocation) (Wallach et al., 2009c; Buntine, 2009). However, the evaluation of topic models that use the bagged representation of text, see Table 2.1 for a summary, has not yet been explored. Thus, the study of the intrinsic evaluation in these models is not only of great importance per se, but it will also enable us later on to compare across different probability models that are based on the bagged representation.

The GaP (Gamma Poisson) model, introduced in Section 2.4.5, and its non-parametric counterparts that build on the NBP (Negative Binomial Process), presented in Section 2.4.6, are forms of factor analysis with Poisson likelihoods, commonly referred to as PFA (Poisson Factor Analysis). Apart from modeling bagged text (Canny, 2004; Zhou et al., 2012),

these models have also been extensively used to model genomic sequences (Zhang et al., 2016b), user ratings (Gopalan et al., 2015) or spatial occurrences (Oliveira, 2013). Furthermore, extensions of PFA suitable for binary data, presented as BPFA (Bernoulli PFA) in Section 2.4.6, have also been used in other fields like network analysis with unweighted edges (Zhou, 2015; Hu et al., 2016). Although the results from this part also apply to a wide variety of count data, we develop the theory and methods around text.

As we have presented in Chapter 2, PFA models build on the bagged representation of text, as opposed to other popular topic models like LDA (Blei et al., 2003) which consider the sequenced representation. Because the document length in the bagged representation is a distribution hyperparameter, PFA models can place a hyperprior on the length and avoid the model to be conditioned on it, as shown in Fig. 2.10 for the GaP model. This does not only provide extra capacity to the model (Canny, 2004), but it also turns it into a fully generative model capable of synthesizing documents in accordance with the topics learned. Despite the popularity of PFA models for a wide variety of tasks, their intrinsic evaluation as probability models, as presented in Section 2.6.1, has not yet been studied. To circumvent this, authors in (Zhou et al., 2012; Zhou and Carin, 2015; Zhou, 2015; Hu et al., 2016) have used a likelihood score that holds out a few words instead of a few documents. However, rigorous studies have not yet been conducted to prove whether this score is well correlated with the intrinsic evaluation of PFA, and hence, a valid metric to compare across different models. In contrast, the intrinsic evaluation of LDA has been studied in far more detail in (Wallach et al., 2009c; Buntine, 2009). While Wallach et al. (2009c) presented different estimation methods for the marginal document likelihood in LDA, Buntine (2009) provided a closed-form expression for this marginal which enabled to compare the accuracy of the existing estimation methods and propose new unbiased estimators. In our opinion, the existence of a closed-form expression allows the assessment of the estimation methods in terms of the accuracy to the true value at least in small setups.

In this chapter, we benefit from the recent finding of a closed-form expression for the marginal likelihood of Poisson factorisation (Filstroff et al., 2018) to develop analytic expressions for the marginal document likelihood in PFA and BPFA from Section 2.4.6. However, due to computational complexity reasons the exact evaluation of PFA is only tractable in small setups with up to 5 topics, documents with up to 10 non-zero words and all words having 1 or 2 counts; and that of BPFA has not even a closed-form expression. Thus, estimation methods are required for approximating the marginal document likelihood in realistic scenarios. Although generic likelihood estimation methods exist in the literature, we show that no previous work has yet considered their use for approximating this likelihood in PFA and BPFA. Therefore, this chapter paves the way to propose and evaluate estimation methods by introducing a rigorous experimental methodology to compare their accuracy and convergence. Furthermore, the study of unbiased likelihood estimation in these models will enable to calibrate other evaluation tasks such as document completion and word prediction.

In what follows, we define in Section 6.1 the intrinsic evaluation of PFA and BPFA as a problem of likelihood estimation. Based on this, we then present the related work for estimation methods as well as for other evaluation strategies. In Section 6.3, we present the experimental setup to assess the accuracy and convergence of likelihood estimation methods in document collections. Finally, we close this chapter by summarizing the main points in Section 6.4.

## 6.1 Problem Definition

In this section, we formalise the problem of intrinsic evaluation for PFA and BPFA as one of likelihood evaluation. As in Section 2.6.1, we use the posterior predictive distribution as a metric to evaluate the generalisation capabilities of a probability model to predict unseen observations. Under this metric, a better model produces higher posterior predictive probabilities on held-out data. That is, the problem can be formally defined for LVM (Latent Variable Model) in general and PFA in particular through the subsequent definitions:

**Problem 6.1.** Given a training set  $\mathbf{x} = \{\mathbf{x}_1, \dots, \mathbf{x}_N\}$ , a held-out set  $\mathbf{x}^* = \{\mathbf{x}_1^*, \dots, \mathbf{x}_{N_t}^*\}$  and the LVM  $\mathcal{M}$ , we seek to compute the posterior predictive distribution  $p(\mathbf{x}^*|\mathbf{x}; \mathcal{M})$ .

Although the training and held-out sets are conditionally independent given the global parameters  $\phi$ , the posterior predictive distribution in Problem 6.1 involves the following integral over these global latent parameters,

$$p(\mathbf{x}^*|\mathbf{x}; \mathcal{M}) = \int p(\mathbf{x}^*|\phi)p(\phi|\mathbf{x}) d\phi. \quad (6.1)$$

However, it is common to consider point estimates for the global latent variables (Walach et al., 2009c; Buntine, 2009) i.e. the mean value or mode of their posterior  $p(\phi|\mathbf{x})$ , and solve instead the following problem:

**Problem 6.2.** Given the point estimates for the global variables  $\hat{\phi}$ , a held-out set  $\mathbf{x}^* = \{\mathbf{x}_1^*, \dots, \mathbf{x}_{N_t}^*\}$ , and the LVM  $\mathcal{M}$ , we seek to compute the likelihood  $p(\mathbf{x}^*|\hat{\phi}; \mathcal{M})$ .

The likelihood above factorises across the  $N_t$  held-out observations for LVMs, but each likelihood still involves a marginal over the local latent variables  $\mathbf{z}^*$ , and hence we need to solve the following  $N_t$  integrals,

$$p(\mathbf{x}^*|\hat{\phi}; \mathcal{M}) = \prod_{n=1}^{N_t} p(\mathbf{x}_n^*|\hat{\phi}) = \prod_{n=1}^{N_t} \int p(\mathbf{x}_n^*, \mathbf{z}_n^*|\hat{\phi}) d\mathbf{z}_n^*. \quad (6.2)$$

where we refer to each integral  $p(\mathbf{x}_n^*|\hat{\phi}; \mathcal{M}) = \int p(\mathbf{x}_n^*, \mathbf{z}_n^*|\hat{\phi}) d\mathbf{z}_n^*$  as the *marginal document likelihood*.

The solution for each marginal ultimately depends on the LVM, hence we next particularize the problem for PFA and BPFA.

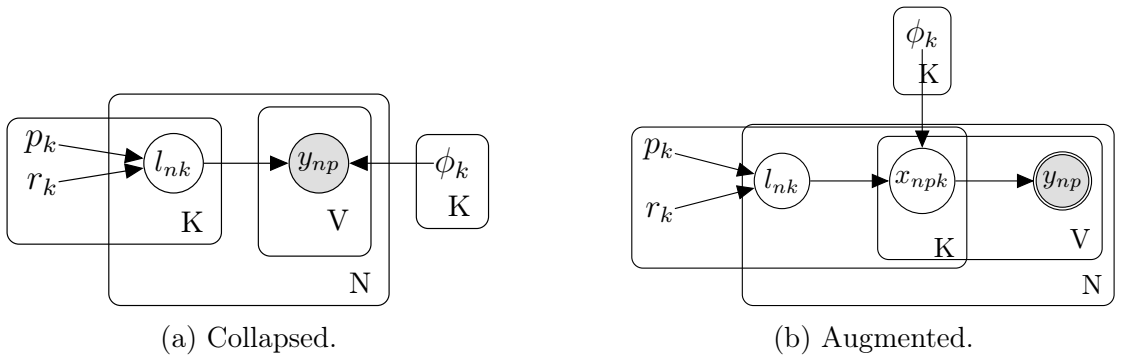


Figure 6.1: PFA graphical models with global point estimates.

Let us first consider the simplified PFA graphical model in Fig. 6.1b, in which the non-parametric model presented before in Fig. 2.12 has been modified with point estimates for the global latent variables  $\{\Phi, \mathbf{p}, \mathbf{r}\}$ . Note that these variables are now drawn as constant values and their parents have been removed from the graph. Furthermore, we know that by collapsing the topic counts  $x$  in Fig. 6.1b, we can obtain the collapsed PFA model in Fig. 6.1a which is equivalent to that in Fig. 2.9. Therefore, we can particularise Problem 6.2 for both PFA models as follows,

**Problem 6.3.** Given the point estimates for the global variables  $\{\Phi, \mathbf{p}, \mathbf{r}\}$  for a PFA model in Fig. 6.1 and a held-out collection of documents  $\mathbf{y}^* = \{y_{1:}^*, \dots, y_{N_t:}^*\}$ , we seek to compute the likelihood  $p(\mathbf{y}^*; \Phi, \mathbf{p}, \mathbf{r}) = \prod_{n=1}^{N_t} p(y_{n:}^*; \Phi, \mathbf{p}, \mathbf{r})$ .

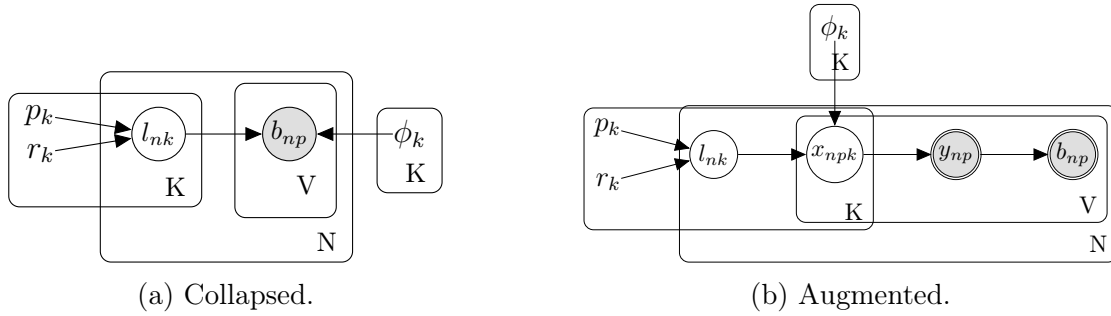


Figure 6.2: BPFA graphical models with global point estimates.

Similarly, one can specify a simplified graphical model for the collapsed and augmented versions of BPFA as depicted by Fig. 6.2a and Fig. 6.2b, respectively. Note that the collapsed version can be defined because the marginal distribution of the observed indicators  $b_{n:}$  given the latent factors  $l_{n:}$  is a Bernoulli distribution given by Eq. (2.34). Thus, we can particularize Problem 6.2 for both BPFA models as follows,

**Problem 6.4.** Given the point estimates for the global variables  $\{\Phi, \mathbf{p}, \mathbf{r}\}$  for a BPFA model in Fig. 6.2 and a held-out collection of binarised documents  $\mathbf{b}^* = \{b_{1:}^*, \dots, b_{N_t:}^*\}$ , we seek to compute the likelihood  $p(\mathbf{b}^*; \Phi, \mathbf{p}, \mathbf{r}) = \prod_{n=1}^{N_t} p(b_{n:}^*; \Phi, \mathbf{p}, \mathbf{r})$ .

In what follows, we derive closed-form and analytic expressions for the marginal document likelihoods in Problem 6.3 and Problem 6.4, respectively. Because the problem is equivalent for training and held-out documents, we omit the distinction between them and derive the above-mentioned expressions for  $p(y_{n:}; \Phi, \mathbf{p}, \mathbf{r})$  and  $p(b_{n:}; \Phi, \mathbf{p}, \mathbf{r})$ .

### 6.1.1 Closed-form Marginal Document Likelihood for PFA

Filstroff et al. (2018) showed that a closed-form marginal likelihood for PFA could be derived from the augmented PFA model in Fig. 6.1b. Firstly, we note that the marginal can be written by making explicit the deterministic relationship between the observed word counts  $y_{n:}$  and latent topic counts  $x_{n:}$ . That is, given that  $y_{n:}$  are sums of  $x_{n:}$  across topics, one can re-express the marginal document distribution as,

$$p(y_{n:}; \Phi, \mathbf{p}, \mathbf{r}) = \sum_{x_{n:} \in \mathbb{X}_{y_{n:}}} \int p(x_{n:}, l_{n:}; \Phi, \mathbf{p}, \mathbf{r}) dl_{n:} \quad (6.3)$$



where the summation set  $\mathbb{X}_{y_n} = \{x_{n::} \in \mathbb{N}_0^{V \times K} \mid y_n = \sum_{k=1}^K x_{n:k}\}$  contains all matrices of non-negative integers  $x_{n::}$  whose rows add up to the word counts of the  $n$ -th document  $y_n$ .

Secondly, we note that the joint distribution on  $x_{n::}$  and  $l_{n::}$  factorises across topics according to the graphical model in Fig. 6.1b and hence,

$$p(y_n; \Phi, \mathbf{p}, \mathbf{r}) = \sum_{x_{n::} \in \mathbb{X}_{y_n}} \prod_{k=1}^K \int p(x_{n:k}, l_{nk}; \phi_k, p_k, r_k) dl_{nk} \quad (6.4)$$

where each marginal  $p(x_{n:k}; r_k, p_k, \phi_k) = \int p(x_{n:k}, l_{nk}; \phi_k, p_k, r_k) dl_{nk}$  can be calculated independently.

Finally, each marginal distribution is the result of compounding  $V$  independent Poisson distributions with a scaled gamma random variable on the rates, as indicated by Proc. 2.9. The compound distribution can then be found by solving the integral,

$$p(x_{n:k}; \phi_k, p_k, r_k) = \int \prod_{p=1}^V \text{Pois}(x_{npk} | l_{nk} \phi_{kp}) \text{Ga}\left(l_{nk}; r_k, \frac{p_k}{1 - p_k}\right) dl_{nk} \quad (6.5)$$

which corresponds to the NM (Negative Multinomial) distribution, as defined in Eq. (B.14), and is parametrised by,

$$p(x_{n:k}; \phi_k, p_k, r_k) = \text{NM}\left(x_{n:k}; r_k, \frac{p_k \phi_k}{1 - p_k + p_k \sum_p \phi_{kp}}\right) \quad (6.6)$$

where  $\phi_k, p_k, r_k$  are the topic-dependent global variables for which we have assumed a point estimate.

In conclusion, the closed-form marginal document likelihood for PFA can be computed exactly by solving,

$$p(y_n; \Phi, \mathbf{p}, \mathbf{r}) = \sum_{x_{n::} \in \mathbb{X}_{y_n}} \prod_{k=1}^K \text{NM}\left(x_{n:k}; r_k, \frac{p_k \phi_k}{1 - p_k + p_k \sum_p \phi_{kp}}\right). \quad (6.7)$$

Next, we present the computational cost to compute the exact marginal given by Eq. (6.7) and discuss its tractability for real document collections and model sizes.

#### 6.1.1.1 On the Complexity of the Closed-form Marginal

Evaluating Eq. (6.7) means summing the  $K$  independent marginals on  $x_{n:k}$  over all elements in the set  $\mathbb{X}_{y_n}$ . As shown in Eq. (6.6), each marginal consists of a NM distribution which has a cost linear with the number words in the vocabulary  $V$  in an unoptimised implementation of NM, or linear with the number of non-zero words in the  $n$ -th document  $V_{c_n}$  when all zero words are jointly evaluated. Therefore, the cost of each summand is linear with both the number of topics  $K$  and the number of non-zeros  $V_{c_n}$ , since  $K$  marginals need to be computed for each summand.

The number of sums in Eq. (6.7) equals the cardinality of the set  $|\mathbb{X}_{y_n}|$ . The cardinality is given by the product of the partitions in each word. The latter consist of the number of partitions of a natural number, i.e.  $y_{np}$ , into  $K$  parts, which is the combinatorial term

of selecting  $K - 1$  objects from a collection of  $y_{np} + K - 1$ . Therefore, the overall number of partitions for document  $n$  is  $\prod_{\{p|y_{np} \neq 0\}} \binom{y_{np} + K - 1}{K - 1}$ , where  $\{w|y_{np} \neq 0\}$  corresponds to the  $V_{c_n}$  non-zeros in the  $n$ -th document.

In the limit, one can show that this set grows exponentially with both the number of topics and the number of non-zeros  $\mathcal{O}((y_{nmax})^{KV_c})$ . We note that the base of the exponent is the maximum word count in the  $n$ -th document  $y_{nmax}$ . Therefore, the cost of summing over the set  $|\mathbb{X}_{y_n}|$  dominates the complexity of evaluating the exact marginal document likelihood. If we compare this complexity with that of LDA, which is given by  $\mathcal{O}(KLK^L)$  where  $L$  refers to the document length (Buntine, 2009), we note that the former grows faster or equal than LDA's. The equality is satisfied when all non-zero words in PFA have one single count.

As a result, the exact evaluation of the marginal document likelihood for PFA is only tractable for quite small problems, such as in models with 5 topics, documents with 10 non-zero words and all words having 1 or 2 counts. However, the existence of this closed-form expression motivates the development of tailored estimation methods and to calibrate their outputs with the true values.

### 6.1.2 Analytic Marginal Document Likelihood for BPFA

Similarly, one can write down the marginal document likelihood for the augmented BPFA model in Fig. 6.2b by making explicit the deterministic relationship between the observed indicators  $b_{n:}$  and the latent word counts  $y_{n:}$ . That is to say, the marginal document likelihood for BPFA can be expressed as,

$$p(b_{n:}; \Phi, \mathbf{p}, \mathbf{r}) = \sum_{y_{n:} \in \mathbb{Y}_{b_{n:}}} \sum_{x_{n:} \in \mathbb{X}_{y_{n:}}} \int p(x_{n:}, l_{n:}; \Phi, \mathbf{p}, \mathbf{r}) dl_{n:} \quad (6.8)$$

where the set  $\mathbb{Y}_{b_{n:}} = \{y_{n:} \in \mathbb{N}_0^V \mid b_{n:} = \mathbf{1}(y_{n:})\}$  contains all non-negative vectors  $y_{n:}$  of length  $V$  whose elements are either 0 when the  $p$ -th word in the vocabulary is absent,  $b_{np} = 0$ , or any positive integer when it is present,  $b_{np} = 1$ . Note that the inner summand corresponds to the marginal document likelihood for PFA and hence it can be computed in closed-form, as presented earlier. As a result, this marginal can be expressed as,

$$p(b_{n:}; \Phi, \mathbf{p}, \mathbf{r}) = \sum_{y_{n:} \in \mathbb{Y}_{b_{n:}}} \sum_{x_{n:} \in \mathbb{X}_{y_{n:}}} \prod_{k=1}^K \text{NM} \left( x_{n:k}; r_k, \frac{p_k \phi_k}{1 - p_k + p_k \sum_p \phi_{kp}} \right) \quad (6.9)$$

where we note that the expression is analytic but has no closed-form due to the summation over the infinite set  $\mathbb{Y}_{b_{n:}}$ . As a result, the exact evaluation of the marginal document likelihood for BPFA is not even tractable for downsized setups and hence, we need to develop estimation methods to approximate it and bounds to sandwich the estimates and evaluate its accuracy. Note that a lower bound could be built from Eq. (6.9) by truncating the infinite sum over  $\mathbb{Y}_{b_{n:}}$  to a finite number of terms, but we would still have to deal with the complexity of the closed-form marginal inside the sum. In Chapter 8, we will explore a different approach based on variational approximations with less computational cost.

Next, we review the existing estimation methods for marginal likelihoods in LVMs and discuss which methods can be extended for the marginal document likelihood estimation in PFA and BPFA.

| Method                           | LDA                                 | PFA        | BPFA       |
|----------------------------------|-------------------------------------|------------|------------|
| Discrete DS                      | Eq. (10) in (Wallach et al., 2009c) | ✗          | ✗          |
| IS-IP $\equiv$ MFI               | Eq. (11) in (Wallach et al., 2009c) | ✗          | ✗          |
| Continuous DS                    | Eq. (12) in (Wallach et al., 2009c) | Eq. (6.11) | Eq. (6.13) |
| Discrete HM                      | Eq. (15) in (Wallach et al., 2009c) | ✗          | ✗          |
| Continuous HM                    | ✗                                   | Eq. (6.14) | Eq. (6.15) |
| AIS                              | Algo. 1 in (Wallach et al., 2009c)  | ✗          | ✗          |
| Chib-style                       | Algo. 2 in (Wallach et al., 2009c)  | ✗          | ✗          |
| Left-to-right Particle Sampler   | Algo. 3 in (Wallach et al., 2009c)  | ✗          | ✗          |
| Left-to-right Sequential Sampler | Section 3.6 in (Buntine, 2009)      | Chapter 7  | Chapter 7  |

Table 6.1: Summary of existing likelihood estimation methods for LDA, PFA and BPFA. ✗ indicates that the method has not been yet extended and/or its extension is not trivial.

## 6.2 Related Work

Wallach et al. (2009c) presented several estimation methods for evaluating LDA in terms of the held-out likelihood. Buntine (2009) also compared the performance of these methods against the exact calculation for the same LDA model. The conclusion of both studies was that simple and commonly-used estimation methods fail to accurately estimate the document likelihood, specially in high-dimensional scenarios. But Wallach’s Left-to-right algorithm was also modified to a Sequential Sampler scheme and proven to be unbiased by Buntine. Given the quick convergences and unbiasedness properties of the Left-to-right Sequential Sampler, it can now be used as a gold standard for estimation in LDA with large number of samples.

To the best of our knowledge, no prior work exists for document likelihood estimation in PFA. However, it is natural to wonder whether LDA methods can be directly applied in PFA. As we have seen previously, the Gamma-Poisson construction differs from that of LDA and the computational cost of the exact marginal document likelihood is much higher. Therefore, existing estimation methods (Wallach et al., 2009c; Buntine, 2009) for LDA have to be amended accordingly. Next, we extend certain methods proposed for LDA and discuss the hindrances for a straightforward extension of the rest. Table 6.1 provides a summary of the existing methods for LDA and references to the parts of this thesis that extend some of the methods for PFA and BPFA.

### 6.2.1 Direct Sampling (DS)

In contrast with LDA, DS (Direct Sampling) or IS (Importance Sampling) with the prior as proposal cannot be formulated over the discrete variables in PFA, because the observed counts  $y_n$  follow a deterministic relationship with the topic counts  $x_n$ . Therefore, DS has to be formulated over the continuous variables  $l_n$  in the collapsed PFA model given in Fig. 6.1a. The marginal document likelihood for the collapsed PFA model is given by,

$$p(y_n; \Phi, \mathbf{p}, \mathbf{r}) = \int p(y_n | l_n; \Phi) p(l_n; \mathbf{r}, \mathbf{p}) dl_n. \quad (6.10)$$

and it can be approximated by Monte Carlo sampling as,

$$p(y_{n:}; \Phi, \mathbf{p}, \mathbf{r}) \approx \frac{1}{S} \sum_{s=1}^S p(y_{n:} | l_{n:}^{(s)}; \Phi, \mathbf{p}, \mathbf{r}) \quad \text{where } l_{n:}^{(s)} \sim p(l_{n:}; \mathbf{r}, \mathbf{p}), \quad (6.11)$$

where the likelihood  $p(y_{n:} | l_{n:}^{(s)}; \Phi, \mathbf{p}, \mathbf{r}) = \prod_{p=1}^V \text{Pois}(y_{np}; l_{n:}^{(s)} \phi_{:k})$  is the product of  $V$  Poisson distributions with rates computed from the product of vector  $l_{n:}^{(s)}$  and the topic matrix  $\Phi$ . Besides, the vector of  $l_{n:}^{(s)}$  is sampled from  $p(l_{n:}; \mathbf{r}, \mathbf{p}) = \prod_{k=1}^K \text{Ga}(l_{n:}; r_k, \frac{p_k}{1-p_k})$  the product of  $K$  Gamma distributions parametrised with shape  $r_k$  and scale  $\frac{p_k}{1-p_k}$ .

Similarly, one can develop the DS method over the continuous variables  $l_{n:}$  for the collapsed BPFA model in Fig. 6.2a. Its marginal document likelihood can be expressed as,

$$p(b_{n:}; \Phi, \mathbf{p}, \mathbf{r}) = \int p(b_{n:} | l_{n:}; \Phi) p(l_{n:}; \mathbf{r}, \mathbf{p}) dl_{n:} \quad (6.12)$$

and the DS sampler formulated as follows,

$$p(b_{n:}; \Phi, \mathbf{p}, \mathbf{r}) \approx \frac{1}{S} \sum_{s=1}^S p(b_{n:} | l_{n:}^{(s)}; \Phi, \mathbf{p}, \mathbf{r}) \quad \text{where } l_{n:}^{(s)} \sim p(l_{n:}; \mathbf{r}, \mathbf{p}), \quad (6.13)$$

where  $l_{n:}^{(s)}$  samples comes from the same  $K$ -variate Gamma distribution than in PFA and the likelihood  $p(b_{n:} | l_{n:}^{(s)}; \Phi, \mathbf{p}, \mathbf{r}) = \prod_{p=1}^V \text{Ber}(b_{np}; 1 - e^{-l_{n:}^{(s)} \phi_{:p}})$  is the product of  $V$  Bernoulli distributions whose probabilities are a function of the product of vector  $l_{n:}^{(s)}$  and the topic matrix  $\Phi$ .

Although these DS estimators are theoretically unbiased, the main caveat is that the proposal distribution, i.e. the prior, ignores the observed word counts and hence, the estimator might require lots of samples to converge for high-dimensional scenarios, i.e.  $K \uparrow \uparrow$ , where the posterior distribution is far from the prior.

### 6.2.2 Harmonic Mean (HM)

An alternative to the DS method is to use samples from the posterior distribution to build an unbiased estimator known as the HM (Harmonic Mean) method (Newton and Raftery, 1994). This estimator, in contrast to that for LDA, cannot be built for the discrete counts either, but only for the continuous factors  $l_{n:}$  for the very same reasons. The HM method for PFA is formulated as,

$$p(y_{n:} | \Phi, \mathbf{p}, \mathbf{r}) \approx \text{HM}(\{p(y_{n:} | l_{n:}^{(s)}; \Phi, \mathbf{p}, \mathbf{r})\}_{s=1}^S) \quad \text{where } l_{n:}^{(s)} \sim p(l_{n:} | y_{n:}, \Phi, \mathbf{p}, \mathbf{r}), \quad (6.14)$$

where  $p(y_{n:} | l_{n:}^{(s)}; \Phi, \mathbf{p}, \mathbf{r})$  is the same likelihood than in Eq. (6.11) and  $l_{n:}^{(s)}$  are samples from the posterior distribution on  $l_{n:}$ .  $\text{HM}(\{\cdot\}_{s=1}^S)$  indicates the harmonic mean across the  $S$  probabilities. As explained in Section 2.5.1, one can sample the posterior via a Gibbs sampling algorithm, that iteratively samples the complete conditionals of the augmented PFA model in Fig. 6.1b, which is conditionally conjugate.

The HM method for BPFA is also formulated over the continuous variables  $l_{n:}$  and uses the likelihood distribution from Eq. (6.13) to calculate an estimate for the marginal document likelihood as follows,

$$p(b_{n:} | \Phi, \mathbf{p}, \mathbf{r}) \approx \text{HM}(\{p(b_{n:} | l_{n:}^{(s)}; \Phi, \mathbf{p}, \mathbf{r})\}_{s=1}^S) \quad \text{where } l_{n:}^{(s)} \sim p(l_{n:} | b_{n:}, \Phi, \mathbf{p}, \mathbf{r}), \quad (6.15)$$

where samples from the posterior on  $l_n$ , can be also drawn through a Gibbs sampling algorithm on the augmented BPFA model in Fig. 6.2b.

Although this method has been used for likelihood estimation in LDA-like topic models (Griffiths and Steyvers, 2004; Wallach, 2006), Newton and Raftery (1994) already expressed some reservations when introducing it due to the non-stable convergence and high variance.

### 6.2.3 Other Sampling Methods

The deterministic relationship between the observed counts  $y_n$  and the latent topic counts  $x_n$  in PFA prevent the trivial extension of many state-of-the-art methods proposed for likelihood estimation in LDA.

Apart from the discrete versions of DS and HM above, the IS-IP (Iterated Pseudo-Counts) proposed in (Wallach et al., 2009c) and shown to correspond with a Mean-Field Importance MFI (Mean Field Importance) sampler in (Buntine, 2009) cannot be directly extended because it is based on sampling the discrete topic assignments. In the same way, the AIS (Annealed Importance Sampling) (Neal, 2001), is based on sampling a series of tempered distributions that transition between the prior and the posterior on the discrete topic assignments. The Chib-style estimators (Murray and Salakhutdinov, 2009) also chooses a “special” set of latent topic assignments and apply a transition operator in these discrete state space. Finally, both left-to-right algorithms also sample the left-hand topic assignments and there is no straightforward extension to PFA.

As a result of this, Chapter 7 will present an extension for the Left-to-right Sequential Sampler to PFA and BPFA and Chapter 8 will develop the idea of the MFI sampler in (Buntine, 2009) for building VIS (Variational Importance Sampling) estimators for PFA and BPFA, but also, more broadly, for other LVMs.

### 6.2.4 Other Evaluation Tasks

Zhou et al. (2012) have evaluated PFA models for topic modelling by computing likelihood or perplexity scores on held out set of random words in the document-term matrix instead of complete documents. However, rigorous studies has not yet been conducted to validate that this task is well correlated with the marginal likelihood and hence, these scores can be misleading when comparing across different models.

A similar approach in LDA-like topic models consists in holding out the second half of a document, while the first half is added to the training data. The evaluation task, known as document completion (Wallach et al., 2009c), consists then in computing the probability of the second half from an empirical estimate of the topic proportion of the document  $\theta_n$ , which has been learned for the half documents added into the training. Although this task is known to be well correlated but biased for LDA, rigorous studies have not yet been conducted for PFA.

Therefore, the study of estimation methods for the marginal document likelihood in PFA will also pave the way for calibrating evaluation tasks and to develop specialized and unbiased sampling methods that approximate these tasks.

## 6.3 Experimental Setup

The development of estimation methods for intrinsic evaluation of PFA models demands for a robust experimental setup in which to evaluate the estimator properties. In particular, we are interested on assessing the accuracy and the convergence speed of every method. While the former is related to the number of samples needed to plateau, the latter is related to whether or not the estimates are biased.

In what follows, we describe a way to determine whether an estimator is biased or not in small scenarios where the exact computation is feasible. Then, we establish how to assess the speed of convergence of an estimation method in small and realistic scenarios. Finally, we present how to build both types of scenarios from publicly available document collections, commonly used in topic modelling.

### 6.3.1 Assessing the Accuracy

We would like to measure the accuracy of an estimator in small and realistic scenarios.

To compare several document probabilities to their exact marginal, we propose to use the KL (Kullback-Leibler) or relative entropy, which is a proper divergence measure for probability distributions. We can interpret it as the number of extra bits added on average per word due to the use of estimated probabilities instead of the exact in decoding a codebook of length the number of evaluated documents. In particular, we compute the KL divergence of,

$$\text{KL}(p, \hat{p}) = \sum_{n=1}^{N+1} p_n \log \frac{\hat{p}_n}{p_n} \quad (6.16)$$

where  $p = \{p_1, \dots, p_N, p_{N+1}\}$  is the set of probabilities for the  $N$  documents plus the probability of any other document  $p_{N+1} = 1 - \sum_{n=1}^N p_n$ . A low KL value means that the estimation method has accurately approximated the exact marginal likelihood.

### 6.3.2 Assessing the Convergence in Realistic Scenarios

To study the convergence in realistic scenarios, we plot the marginal document likelihood for all documents as function of the number of samples. The log of this likelihood enables to visually analyse and compare the speed of convergence of different methods. The faster this curve reaches the plateau, the better the convergence.

### 6.3.3 Document Collections

Finally, Table 6.2 contains the 6 collections that we propose to use for the experimentation. Note that these collections are ordered decreasingly on the average document length, being data sets at the top commonly used as long text representatives, while those at the bottom are commonly used for short text studies.

| Data set              | Vocabulary | Documents | Document Length |
|-----------------------|------------|-----------|-----------------|
| NIPS Proceedings      | 11,463     | 5,811     | 1,899 $\pm$ 513 |
| Associated Press (AP) | 10,473     | 2,246     | 194 $\pm$ 111   |
| 20Newsgroups (20NGs)  | 11,928     | 18,846    | 123 $\pm$ 247   |
| Reuters               | 8,843      | 19,043    | 79 $\pm$ 75     |
| Twitter               | 6,344      | 10,523    | 25 $\pm$ 4      |
| Web Snippets (WS)     | 4,679      | 12,309    | 9 $\pm$ 3       |

Table 6.2: Document collections.

All data sets, except NIPS which was used as it is published<sup>1</sup>, were pre-processed by removing stopwords, non-letters and words with two or less characters. We have also applied Porter Stemming and filtered out words that appeared less than 5 times or in more than 50% of documents. The processed data sets can be freely access in [Capdevila \(2018\)](#).

To build data sets for tractable scenarios, vocabularies were cropped to the 100 most frequent words and only those documents that lead to a number of partitions lower than  $10^9$  in a model with  $K = 5$  topics were kept.

Finally, to build the binarised data sets, word counts in the pre-processed documents are simply encoded as “1” if the counts are non-zero and as “0” if they are zero.

## 6.4 Summary and Conclusion

In this chapter, we discussed the intrinsic evaluation of PFA models, like GaP, its non-parametric counterparts (i.e. PFA) and its binary extensions (i.e. BPFA). As presented earlier in Section 2.6.1, the intrinsic evaluation of probability models can be conducted through the posterior predictive probability of the held-out documents. We showed that one can similarly evaluate PFA models by approximating this probability taking point estimates of the global variables and then marginalizing out the local variables in the resulting joint distribution of the observed and latent variables. Although there exists a closed-form expression for this marginal in PFA (but not for BPFA), we showed that its calculation is only tractable for reasonably small setups, such as models with up to 5 topics and documents with up to 10 non-zero words whose counts are all less than 3. Therefore, we highlighted the need to approximate this intractable marginal in an unbiased manner for more realistic scenarios, in order to use the predictive probability on the held-out data as an intrinsic evaluation metric useful for comparing different models.

The existing literature has addressed the same problem for the LDA model, but, to our best knowledge, no prior work exists for PFA. Moreover, we showed that the problem is far more intractable in PFA than in LDA and the same estimation methods might not directly apply. As a result, in this chapter we extended simple estimation methods (DS and HM) and we laid down the experimental setup to compare the accuracy and convergence properties of different estimators in tractable and intractable scenarios. In the next chapters, we will address the estimation problem by extending the state-of-the art method in LDA and proposing novel approaches to approximate this marginal. Furthermore, we will conduct

<sup>1</sup><https://archive.ics.uci.edu/ml/datasets/NIPS+Conference+Papers+1987-2015>

thorough experimentation and study the accuracy and convergence of these estimators in small and large scenarios.



# 7

## Left-to-right Sequential Samplers

*“Divide et impera”*

PHILIP II of Macedon

Capdevila, J., Cerquides, J., Torres, J., Petitjean, F., and Buntine, W. (2018c). A left-to-right algorithm for likelihood estimation in gamma-poisson factor analysis. In *Joint European Conference on Machine Learning and Knowledge Discovery in Databases*, pages 638–654. Springer

In the previous chapter, we showed that computing the probability of a single document in PFA (Poisson Factor Analysis) requires integrating out all local latent variables. By means of the augmented PFA model in Fig. 6.1b, we derived a closed-form expression for this marginal document likelihood, whose complexity grows exponentially with the number of topics and the number of non-zero words in the document. Furthermore, the base of the exponential depends on the maximum count of any word in the document. This means that the computation of the exact marginal is only feasible in reasonably small setups. Thus, approximation methods to the marginal document likelihood are essential for evaluating PFA under more realistic conditions.

Simple approximation methods, such as DS (Direct Sampling) or the HM (Harmonic Mean) method (Newton and Raftery, 1994), are known to produce inaccurate estimations, particularly in high-dimensional setups. Despite this, their ease of implementation and low computational cost have encouraged their use in LDA-like models (Griffiths and Steyvers, 2004; Wallach, 2006). Against this background, there is a need for more accurate and computationally efficient estimation methods. One approach which has been reported to produce state-of-the-art results in LDA (Latent Dirichlet Allocation) is the Left-to-right Sequential Sampler (Buntine, 2009). By leveraging on the chain rule of probability, the

algorithm decomposes the joint document probability into a product of conditionals, one conditional per word. Then, unbiased estimates can be built for each conditional given the posterior samples on the left-hand topics.

However, three issues arise due to the Gamma-Poisson construction in PFA:

1. The posterior distribution over the left-hand topic counts is not tractable.
2. The computational cost of each conditional is exponential in the number of topics.
3. The time complexity grows quadratically in the number of non-zero words.

In this chapter, we propose a left-to-right sequential sampler for PFA called L2R that addresses (1) by means of Gibbs sampling on the augmented model, (2) via Importance Sampling with proposal distributions that condition on the left-hand samples (3) through a mathematical simplification that enables computing the conditional probability for all zero words at once. For the sake of comparison, we also introduce the vanilla L2R, a left-to-right sequential sampler that computes the exact condition of (2) in small setups.

Moreover, we extend the L2R sampler for the BPFA (Bernoulli PFA) model whose marginal document likelihood can be computed from that of PFA, as indicated by Eq. (6.9). Despite the fact that we show that this model enables the computation of the exact conditionals in (2), the left-hand topics in (1) are not constrained to sum up to the observed counts but to some latent counts which can be any non-negative integer. As a result, we note that the sampling space of the left-hand counts in BPFA is not finite and it entails the probability mass to spread over larger regions than in PFA. Due to these differences, it is important to empirically validate the performance of the BPFA estimator in short and long binarised text too.

Therefore, we compare the accuracy of L2R to that of DS and HM methods,

- for PFA in reasonably small setups, where the exact marginal can be assessed in moderate time and hence, conclusions about their accuracy can be drawn;
- for PFA and BPFA in realistic scenarios, where the exact marginal and the vanilla left-to-right are computationally infeasible and hence, only their convergence can be studied.

In the rest of this chapter, we first describe the L2R algorithm for PFA in Section 7.1 where we address the three above-mentioned issues. Then, we extend the L2R sampler for the BPFA model and we show the main differences in sampling the left-hand counts and computing the exact conditionals. Section 7.3 contains the experiments carried out in both scenarios for both models. We conclude this chapter in Section 7.4 summarising the main contributions and pointing at open problems.

## 7.1 A Left-to-right Sampler for PFA

In this section we present L2R, a left-to-right sequential sampler for PFA. L2R builds on the general product rule of probability, in which any joint distribution can be decomposed into the product of several conditionals. By considering a left-to-right order of words, the joint probability of a document is decomposed by the product of  $V$  conditional probabilities

where each is conditioned on the preceding left words. We can express this decomposition for PFA as,

$$p(y_{n:}; \Phi, \mathbf{p}, \mathbf{r}) = \prod_{p=1}^V p(y_{np} | y_{n<p}; \Phi, \mathbf{p}, \mathbf{r}) \quad (7.1)$$

where " $< p$ " refers to words on the left side of  $p$ . Nonetheless, the exact calculation of these conditionals is still intractable. We now introduce the left-hand topic counts  $x_{n<p:}$  and marginalize them out as follows,

$$p(y_{n:}; \Phi, \mathbf{p}, \mathbf{r}) = \prod_{p=1}^V \sum_{x_{n<p:}} p(y_{np}, x_{n<p:} | y_{n<p}; \Phi, \mathbf{p}, \mathbf{r}). \quad (7.2)$$

Given that the  $p$ -th word count,  $y_{np}$ , is conditionally independent from the left-hand counts  $y_{n<p}$  given their topic counts  $x_{n<p:}$ , the joint probability above can be split into two factors as,

$$p(y_{n:}; \Phi, \mathbf{p}, \mathbf{r}) = \prod_{p=1}^V \sum_{x_{n<p:}} p(y_{np} | x_{n<p:}; \Phi, \mathbf{p}, \mathbf{r}) p(x_{n<p:} | y_{n<p}; \Phi, \mathbf{p}, \mathbf{r}). \quad (7.3)$$

This expression uncovers a sampling structure which suggests to draw samples from the posterior over the topic counts on the left-hand side of  $p$  and to evaluate the conditional probability of the current word count given these left-hand samples. In other words, the two step process can be summarised as follows,

$$x_{n<p:}^{(s)} \sim p(x_{n<p:} | y_{n<p}; \Phi, \mathbf{p}, \mathbf{r}) \quad (7.4)$$

$$p(y_{n:}; \Phi, \mathbf{p}, \mathbf{r}) \approx \prod_{p=1}^V \frac{1}{S} \sum_{s=1}^S p(y_{np} | x_{n<p:}^{(s)}; \Phi, \mathbf{p}, \mathbf{r}) \quad (7.5)$$

or expressed through a simplified Left-to-right sequential sampler depicted in Algorithm 7.1. Given  $S$  samples, the  $n$ -th document  $y_{n:}$  and the global point estimates  $\{\Phi, \mathbf{p}, \mathbf{r}\}$ , the algorithm loops though the  $V$  words in the vocabulary and for each word compute its conditional probability on the left-hand word counts (line 5). Each conditional is then approximated via Monte Carlo sampling by drawing  $S$  samples from the posterior distribution over the left-hand topic counts (line 3) and evaluating its likelihood (line 4). Finally, the marginal document likelihood is computed by multiplying the  $V$  unbiased estimations of the conditionals.

---

**Algorithm 7.1:** Simplified pseudocode for L2R algorithm.

---

**input** :  $S, y_{n:}, \Phi, \mathbf{p}, \mathbf{r}$   
**output:**  $p(y_{n:}; \Phi, \mathbf{p}, \mathbf{r})$

```

1 for  $p \leftarrow 1$  to  $V$  do
2   for  $s \leftarrow 1$  to  $S$  do
3      $x_{n<p:}^{(s)} \sim p(x_{n<p:} | y_{n<p}; \Phi, \mathbf{p}, \mathbf{r})$ 
4      $p(y_{np} | x_{n<p:}^{(s)}; \Phi, \mathbf{p}, \mathbf{r}) \leftarrow p(y_{np} | x_{n<p:}^{(s)}; \Phi, \mathbf{p}, \mathbf{r})$ 
5    $p(y_{np} | y_{n<p}; \Phi, \mathbf{p}, \mathbf{r}) = \frac{1}{S} \sum_s p(y_{np} | x_{n<p:}^{(s)}; \Phi, \mathbf{p}, \mathbf{r})$ 
6  $p(y_{n:}; \Phi, \mathbf{p}, \mathbf{r}) \approx \prod_{p \leq V} p(y_{np} | y_{n<p}; \Phi, \mathbf{p}, \mathbf{r})$ 

```

---

Next, we present a method for drawing samples from the posterior on the left-hand topic counts in Eq. (7.4) and a strategy to approximate the inner conditionals in Eq. (7.5). This will enable us to address the first two issues mentioned in the introduction. Then, we show that by re-ordering documents in a particular way, we can compute the product in Eq. (7.5) across the non-zero words in the  $n$ -th document  $V_{c_n}$ , which addresses the third issue. Finally, we summarise all these contributions in the pseudo-code for the L2R algorithm and discuss its computational complexity.

### 7.1.1 Sampling the Left-hand Topics from Word Counts

The posterior distribution in Eq. (7.4) does not have a closed-form expression due to the intractable normalising constant. Therefore, a common thing to do is to build a Gibbs sampler to draw samples from it. However, the complete conditionals  $p(x_{np'} | x_{n<p}^{-p'}, y_{n<p}; \Phi, \mathbf{p}, \mathbf{r}) \forall p' < p$  do not admit a computationally feasible sampler due to the conditioning on the observed counts  $y_{np'}$ .

One way to sample from this posterior is to consider the augmented PFA model in Fig. 6.1b, but only over the left-hand side of  $p$ . This augmentation makes the model locally conjugate and it enables the derivation of the complete conditionals as,

$$p(l_{nk} | -) = \text{Ga} \left( l_{nk}; r_k + \sum_{p' < p} x_{n<p'k}, \frac{p_k}{1 - p_k + p_k \sum_{p' < p} \phi_{kp'}} \right) \quad \forall k \leq K \quad (7.6)$$

$$p(x_{np'} | -) = \text{Mult} \left( x_{np'}; y_{np'}, \frac{\phi_{:p'} l_{n:}}{\sum_k \phi_{kp'} l_{nk}} \right) \quad \forall p' < p \quad (7.7)$$

where “ $| -$ ” refers to all variables except the conditioned. These expressions can be integrated in a Gibbs sampling scheme, as explained in Section 2.5.1, in which we first sample Eq. (7.6) and then each of the left word counts as in Eq. (7.7), or vice-versa. However, only samples from the left-hand topics need to be recorded for the L2R algorithm.

### 7.1.2 Approximating the Conditional Probability

The inner conditional probability in Eq. (7.5) can be expressed as the sum of the marginal on  $x_{np}$ : over all possible topic counts, which must add up to the  $p$ -th word count  $y_{np}$ . Given that topic counts are independent among them, the marginal also factorises. We can write this as,

$$p(y_{np} | x_{n<p}^{(s)}; \Phi, \mathbf{p}, \mathbf{r}) = \sum_{x_{npk} \in \mathbb{X}_{y_{np}}} \prod_{k=1}^K p(x_{npk} | x_{n<p}^{(s)}; \phi_{k:}, p_k, r_k). \quad (7.8)$$

where the summation set  $\mathbb{X}_{y_{np}} = \{x_{np} \in (\mathbb{N} \cup 0)^K \mid y_{n:} = \sum_{k=1}^K x_{n:k}\}$  has cardinality  $|\mathbb{X}_{y_{np}}| = \binom{y_{np} + K - 1}{K - 1}$ .

The marginal above, which is conditioned on the left samples, can be derived by leveraging on the augmented model. By introducing  $l_{nk}$ , the probability of the actual count  $x_{npk}$  becomes conditionally independent of the left samples  $x_{n<p}^{(s)}$  given the introduced  $l_{nk}$ . Therefore, the left samples influence the probability over  $l_{nk}$ , but not that over  $x_{npk}$  as shown,

$$p(x_{npk}|-) = \int p(x_{npk}|l_{nk}; \phi_{kp}) p(l_{nk}|x_{n<pk}^{(s)}; \phi_{k:}, p_k, r_k) dl_{nk} \quad (7.9)$$

where " $-$ " refers to the set  $\{x_{n<pk}^{(s)}, \phi_{k:}, p_k, r_k\}$ .

In the integral above, we substitute the probability over  $x_{npk}$  for  $\text{Pois}(x_{npk}; l_{nk}\phi_{kp})$  and that over  $l_{nk}$  for the complete conditional in Eq. (7.6). The resulting integral corresponds to a NB (Negative Binomial) <sup>1</sup> parameterized as follows,

$$p(x_{npk}|-) = \text{NB} \left( x_{npk}; r_k + \sum_{p'<p} x_{np'k}^{(s)}, \frac{\phi_{pk}p_k}{1 - p_k + p_k \sum_{p'\leq p} \phi_{p'k}} \right). \quad (7.10)$$

Although it is possible to compute the exact conditional probability through the closed-form expression given by Eq. (7.8), its computational cost still grows exponentially with the number of topics (note that the exponential growth is now independent of the number of non-zeros) and hence it is only tractable for a small number of topics or word counts  $y_{np}$ .

Therefore, our alternative to the exact calculation consists in replacing the complicated sum in Eq. (7.8) with a Monte Carlo estimate. To do that, we propose to perform Importance Sampling, described in Section 2.5.3, with a proposal distribution which is conditioned on the left samples as follows,

$$Q(x_{np:}|x_{n<p:}^{(s)}; \phi_{:p}, \mathbf{p}, \mathbf{r}) = \text{Mult}(x_{np:}; y_{np}, \propto \phi_{:p} \mathbb{E}_{p(l_{n:}|x_{n<p:}^{(s)}, \Phi, \mathbf{p}, \mathbf{r})} [l_{n:}]) \quad (7.11)$$

where expectation over  $l_{n:}$  is computed w.r.t the complete conditional in Eq. (7.6). Given that this proposal is built taking into account the left-hand samples, the proposal will be close to the marginal  $x_{np:}$  as long as the left counts are good predictors of the target.

Finally, we estimate the conditional probability as,

$$\begin{aligned} x_{np:}^{(s')} &\sim Q(x_{np:}|x_{n<p:}^{(s)}; \phi_{:p}, \mathbf{p}, \mathbf{r}) \\ p(y_{np}|x_{n<p:}^{(s)}; \Phi, \mathbf{p}, \mathbf{r}) &\approx \frac{1}{S'} \sum_{s'} \frac{p(x_{np:}^{(s')}|x_{n<p:}^{(s)}; \Phi, \mathbf{p}, \mathbf{r})}{Q(x_{np:}^{(s')}|x_{n<p:}^{(s)}; \phi_{:p}, \mathbf{p}, \mathbf{r})} \end{aligned} \quad (7.12)$$

where  $S'$  corresponds to another set of samples which replace the intractable sum in Eq. (7.8). However, we will show in the experiments that with one single sample  $S' = 1$ , we can provide an accurate approximation in situations where the topics for the  $p$ -th word are likely to be predicted from the preceding topics, which is often the case if some thematic structure exists in the corpus.

### 7.1.3 Dealing with Zero Words

The left-to-right decomposition rule in Eq. (7.1) does not impose any specific word order to be valid. Besides, the inspection of the exact conditional formula from Eqs. (7.8) (7.10) reveals that words without counts contribute with a tractable term which only depends on the left-hand counts.

This suggests that if we reorder documents in such a way that all non-zero words precede zeros, we can reuse the posterior samples drawn for non-zero words to calculate the

---

<sup>1</sup>the NB probability distribution is given in Eq. (B.12)

probability of zeros. Note that zeros do not contribute to the posterior sampling over the left-hand topics. This allows to build a conditional probability for all words without counts  $p \geq v_z$  that occur after the non-zeros  $p < v_z$ . A closed-form expression can be derived for this probability which can be computed in linear time with the number of topics as,

$$p(y_{n \geq v_z} | x_{n < v_z}^{(s)}; \Phi, \mathbf{p}, \mathbf{r}) = \prod_k \left( \frac{1 - p_k + p_k \sum_{p < v_z} \phi_{pk}}{1 - p_k + p_k \sum_{p \leq V} \phi_{pk}} \right)^{r_k + \sum_{p < v_z} x_{npk}^{(s)}}. \quad (7.13)$$

By re-ordering the document, reusing the posterior samples and the mathematical simplification shown above, we can speed up the algorithm from computing the conditional probability across all words in the vocabulary  $V$  to only those with non-zero counts  $V_{c_n}$ . Given that for most corpora, the vocabulary size is widely larger than the non-zero words per document ( $V \gg V_{c_n}$ ), this makes a critical enhancement to the time-complexity of this algorithm as we show later.

### 7.1.4 Algorithm Pseudocode

In Algorithm 7.2, we present the pseudocode of L2R, summarising the developments from the previous sections. The input data consists of the number of samples  $S$  used to approximate each of the factors in the left-to-right decomposition, the number of samples  $S'$  to draw from the proposal distribution in the case of sampled conditionals, the  $n$ -th document  $y_n$ : sorted as in Section 7.1.3 and the point estimates for the global parameters  $\Omega = \{\Phi, \mathbf{p}, \mathbf{r}\}$ . The algorithm outputs the approximate marginal document likelihood  $p(y_n; \Phi, \mathbf{p}, \mathbf{r})$ .

---

**Algorithm 7.2:** Pseudocode for L2R algorithm.

---

```

input  :  $S, S', y_n, \Omega = \{\Phi, \mathbf{p}, \mathbf{r}\}$ 
output:  $p(y_n; \Omega)$ 

1 for  $p \leftarrow 1$  to  $V_{c_n}$  do
2   for  $s \leftarrow 1$  to  $S$  do
3      $x_{n < p}^{(s)} \leftarrow \text{PostSamp}(x_{n < p}^{(s)}, \Omega);$  Eqs. (7.6) (7.7)
4      $p(y_{np} | x_{n < p}^{(s)}; \Omega) \leftarrow \text{CondProb}(x_{n < p}^{(s)}, \Omega, S');$  Eq. (7.12)
5      $p(y_{np} | y_{n < p}; \Omega) = \frac{1}{S} \sum_s p(y_{np} | x_{n < p}^{(s)}; \Omega)$ 
6  $v_z \leftarrow V_{c_n} + 1$ 
7 for  $s \leftarrow 1$  to  $S$  do
8    $x_{n < v_z}^{(s)} \leftarrow \text{PostSamp}(x_{n < v_z}^{(s)}, \Omega);$  Eqs. (7.6) (7.7)
9    $p(y_{n \geq v_z} | x_{n < v_z}^{(s)}; \Omega) \leftarrow \text{CondProbZeros}(x_{n < v_z}^{(s)}, \Omega);$  Eq. (7.13)
10  $p(y_{nv_z} | y_{n < v_z}; \Omega) = \frac{1}{S} \sum_s p(y_{n \geq v_z} | x_{n < v_z}^{(s)}; \Omega)$ 
11  $p(y_n; \Omega) \approx \prod_{p \leq v_z} p(y_{np} | y_{n < p}; \Omega)$ 

```

---

From line 1 to 5, the algorithm approximates the conditional distributions for non-zero words by computing the averaged probability across  $S$  samples for each word. To approximate this conditional probability, the algorithm uses the Importance Sampling scheme defined in Eq. (7.12).

From line 6 to 10, the algorithm approximates the conditionals for all words without counts following the same procedure as for non-zeros, except that the conditional for all non-zeros is computed at once in line 9 through its exact form given by Eq. (7.13).

The final estimate for marginal document likelihood is built from the product of the  $V_{c_n} + 1$  probabilities in line 11.

### 7.1.5 On the Time Complexity of the L2R Algorithm

The time complexity of the L2R algorithm can be derived from the cost of the subprocesses of Algorithm 7.2. We first note that the cost of computing the conditionals for all non-zero words dominates over that of zeros because line 4 is linear in both the number of samples  $S'$  and the number of topics  $K$ , whereas line 9 is only linear in the latter. The cost of the posterior sampling process in line 3 and 8 is also linear in the number of topics  $K$  and the number of non-zero words in the  $n$ -th document  $V_{c_n}$ . Therefore, the overall cost is given by  $\mathcal{O}(SV_{c_n}(V_{c_n} + K + S'))$  which is quadratic in the number of non-zero words. Note also that without the optimization of zeros it would have been quadratic in the vocabulary size and without the approximate conditionals, exponential in the number of topics.

## 7.2 A Left-to-right Sampler for BPFA

The L2R sampler can be extended for the BPFA model following the same process than in Section 7.1. We next repeat these developments for BPFA to show that the sampler has the same structure but different ways to compute the left-hand topics and its conditionals.

We start with the chain rule of probability that enables to decompose the marginal document likelihood from left to right into  $V$  conditional probabilities, one probability for each word in the vocabulary conditioned on their left-hand words. That is to say,

$$p(b_{n:}; \Phi, \mathbf{p}, \mathbf{r}) = \prod_{p=1}^V p(b_{np} | b_{n<p}; \Phi, \mathbf{p}, \mathbf{r}) \quad (7.14)$$

where  $b_{n:}$  to indicate the binary vector of size  $V$  that indicates the presence or absence of each word in the  $n$ -th document. Similar to PFA,  $b_{np}$  is the indicator for the  $p$ -th word in the  $n$ -th document and  $b_{n<p}$  are the indicators on the left-hand side of  $p$ . As defined in Section 6.1,  $\Phi, \mathbf{p}, \mathbf{r}$  are the point estimates for the global variables of BPFA.

As in Eq. (7.2), we introduce the left-hand topic counts  $x_{n<p}$  and marginalize them out,

$$p(b_{n:}; \Phi, \mathbf{p}, \mathbf{r}) = \prod_{p=1}^V \sum_{x_{n<p}} p(b_{np}, x_{n<p} | b_{n<p}; \Phi, \mathbf{p}, \mathbf{r}). \quad (7.15)$$

Finally, the indicator of the  $p$ -th word,  $b_{np}$ , is conditionally independent from the left-hand indicators,  $b_{n<p}$  given these left-hand counts  $x_{n<p}$ , and hence, we can split the joint distribution above in,

$$p(b_{n:}; \Phi, \mathbf{p}, \mathbf{r}) = \prod_{p=1}^V \sum_{x_{n<p}} p(b_{np} | x_{n<p}; \Phi, \mathbf{p}, \mathbf{r}) p(x_{n<p} | b_{n<p}; \Phi, \mathbf{p}, \mathbf{r}). \quad (7.16)$$

This leads to the same sampler structure than in PFA, but we now have a document as a vector of indicators  $b_{n:}$  instead of a vector of counts  $y_{n:}$ . That is,

$$x_{n<p:}^{(s)} \sim p(x_{n<p:} | b_{n<p:}; \Phi, \mathbf{p}, \mathbf{r}) \quad (7.17)$$

$$p(b_{n:}; \Phi, p, r) \approx \prod_{p=1}^V \frac{1}{S} \sum_{s=1}^S p(b_{np} | x_{n<p:}^{(s)}; \Phi, \mathbf{p}, \mathbf{r}) \quad (7.18)$$

where the differences with the PFA sampler are in the posterior probability on the left-hand topics, Eq. (7.17), and in the conditional probability, Eq. (7.18). We next review how we can sample the left-hand topic counts from the left-hand indicators, as well as how to calculate the conditional probability of the  $p$ -th indicator in the  $n$ -th document.

### 7.2.1 Sampling the Left-hand Topics from Word Indicators

Similar to PFA, the posterior distribution in Eq. (7.17) does not have a parametric distribution easy to sample from. Therefore, one can draw  $x_{n<p:}$  samples from the augmented BPFA model which includes the latent factors  $l_{n:}$  and counts  $y_{n:}$ . Through this augmentation, one can derive the corresponding complete conditionals and integrate them in the Gibbs sampling scheme. The complete conditionals are given by,

$$p(l_{nk} | -) = \text{Ga} \left( l_{nk}; r_k + \sum_{w' < w} x_{n<w'k}, \frac{p_k}{1 - p_k + p_k \sum_{w' < w} \phi_{kw'}} \right) \quad \forall k \leq K \quad (7.19)$$

$$p(x_{np'} | -) = \text{Mult} \left( x_{np'}; y_{np'}, \frac{\phi_{:p'} l_{n:}}{\sum_k \phi_{kp'} l_{nk}} \right) \quad \forall p' < p \quad (7.20)$$

$$p(y_{np'} | -) = \begin{cases} 0, & \text{if } b_{np'} = 0 \\ \text{Pois}_+ (y_{np'}; \sum_k \phi_{kp'} l_{nk}), & \text{if } b_{np'} = 1 \end{cases} \quad \forall p' < p \quad (7.21)$$

where Eq. (7.19) corresponds to the same  $K$  Gamma distributions conditioned to the left-hand topic counts than in Eq (7.6). Likewise, Eq. (7.20) samples the left hand  $p' < p$  topic counts from Multinomials conditioned to the  $l_{n:}$  factors and word counts  $y_{np'}$ . However, the word counts  $y_{np'}$  are now latent and Eq. (7.21) establishes how to sample them. It does it from a zero-truncated Poisson distribution if the word is present,  $b_{np'} = 1$ , or it forces the counts  $y_{np'}$  to be zero if the word is absent,  $b_{np'} = 0$ . Again, only samples from the left-hand topics  $x_{n<p:}$  have to be recorded for the L2R algorithm in BPFA.

We note that the sampling of the counts for each word increases vastly the sampling space of BPFA in comparison to that of PFA. Thus, we expect that the sampling of this extra count variable to impact negatively the convergence of this estimator, specially for long documents (with lots of present words). However, we need to quantify empirically how badly this extra sampling compromises the estimation.

### 7.2.2 Computing the Conditional Probability

The conditional probability in Eq. (7.18) can be expressed as the marginals over the latent variables  $x_{np:}$  and  $y_{np}$  of their conditional on the left-hand samples. We can write this as,

$$p(b_{np} | x_{n<p:}^{(s)}; \Phi, \mathbf{p}, \mathbf{r}) = \sum_{y_{np} \in \mathbb{Y}_{b_{np}}} \sum_{x_{npk} \in \mathbb{X}_{y_{np}}} \prod_{k=1}^K p(x_{npk} | x_{n<p:}^{(s)}; \phi_{k:}, p_k, r_k) \quad (7.22)$$



where  $\mathbb{X}_{y_{np}}$  is the same summation set than in Eq. (7.8) and  $\mathbb{Y}_{b_{np}} = \{y_{np} \in \mathbb{N}_0 \mid b_{np} = \mathbf{1}(y_{np})\}$ . Because of this double marginal and the fact that  $\mathbb{Y}_{b_{np}}$  is not finite, the evaluation of this conditional seems far more intractable than in PFA, but fortunately the binary states of  $b_{np}$  enable to exactly compute this conditional with very little cost. We first note that an absent word  $b_{np} = 0$  reduces the set to  $\mathbb{Y}_{b_{np}} = \{0\}$  and hence, the set  $\mathbb{X}_{y_{np}} = \{(0, \dots, 0)\}$ . As a result, the probability for an absent word is basically the product of the  $K$  probabilities, each evaluated at 0. Second, the probability of a present word is simply one minus the probability of the absent word, given that both must add up to 1. Mathematically, that is to say,

$$p(b_{np} | x_{n < p}; \Phi, \mathbf{p}, \mathbf{r}) = \begin{cases} \prod_{k=1}^K p(x_{npk} = 0 | x_{n < pk}^{(s)}; \phi_{k:}, p_k, r_k), & \text{if } b_{np} = 0 \\ 1 - \prod_{k=1}^K p(x_{npk} = 0 | x_{n < pk}^{(s)}; \phi_{k:}, p_k, r_k), & \text{if } b_{np} = 1 \end{cases} \quad (7.23)$$

where  $p(x_{npk} = 0 | x_{n < pk}^{(s)}; \phi_{k:}, p_k, r_k)$  is given by the same NB distribution in Eq. (7.10).

Finally, we note that the posterior sampling in the previous section and the exact computation of the conditional probability can be integrated in the same Algorithm 7.2 by replacing Eqs. (7.6) (7.7) by Eqs. (7.19) (7.20) (7.21) in lines 3 and 8 and Eq. (7.12) by Eq. (7.23) in line 4. Because, one can deal with all absent words equally than with zero words in PFA, the probability for all zero words in Eq. (7.13) can be still used for the BPFA L2R. Therefore, the time complexity for this algorithm is  $\mathcal{O}(SV_{c_n}(V_{c_n} + K))$ , which is faster than that for PFA since it does not require to draw the  $S'$  importance samples, but it is still quadratic with the number of absent words in the  $n$ -th document  $V_{c_n}$ .

## 7.3 Empirical Evaluation

In this section, we present the comparison of the proposed L2R algorithms against DS and HM methods for PFA and BPFA. Following the experimental setup described in Section 6.3, we evaluate the accuracy of the three PFA samplers by comparing their estimates to the exact marginal likelihood in reasonably small setups and we also study the convergence properties of these methods for PFA and BPFA in more realistic scenarios. The lack of a closed-form expression for the marginal of BPFA hampers the comparison of the BPFA samplers with the exact. The code for all estimation methods has been made public<sup>2</sup>, the document collections have been presented in Section 6.3.3 and the configuration parameters for training the corresponding models as well as for setting up the samplers are presented next.

### 7.3.1 Model Training and Samplers Settings

Among all possible PFA models, we have chosen to train the  $\beta\Gamma$ -PFA model (Zhou et al., 2012), which considers the non-marked Beta Process on  $\mathbf{p}$ , so it does not infer the  $\mathbf{r}$  variables as in  $\beta\gamma\Gamma$ -PFA. The motivation to choose  $\beta\Gamma$ -PFA is that it corresponds to the non-parametric version of GaP (Gamma Poisson) (Canny, 2004) which was proposed as an alternative to LDA (Blei et al., 2003) and it is considered to be the most basic Poisson factor analysis model. Besides, the non-parametric prior of  $\beta\Gamma$ -PFA allows us to avoid model

<sup>2</sup><https://github.com/jcapde/L2R>

selection on a critical parameter such the number of topics. Despite the choice of  $\beta\Gamma$ -PFA, the estimators developed in this chapter are valid for  $\beta\gamma\Gamma$ -PFA and many other PFA models. For estimation purposes, these two models only differ on the global parameters, and hence, we expect to reach the same conclusions if  $\beta\gamma\Gamma$ -PFA would have been used.

The hyperparameters of the  $\beta\Gamma$ -PFA are set according to (Zhou et al., 2012). We use a symmetric Dirichlet prior  $\eta = 0.1$  on the topic distributions  $\Phi$ . The finite Beta Process uses  $\epsilon = 0.01$  and  $c = 1$ . Besides, the shape hyperparameters on the Gamma-distributed factors  $\mathbf{r}$  are set all to 1. Then, the training of this model is performed through the Gibbs Sampling algorithm described in (Zhou et al., 2012). We decide to run the algorithm for 1,000 iterations and to discard a burn-in period of 500 samples. The algorithm also uses an upper-bound for the number of topics  $K_{max}$  which is set to  $\epsilon^{-1} = 100$  for realistic document collections and  $K_{max} = 5$  (and hence,  $\epsilon = 0.2$ ) for downsized collections. The point estimates for the global parameters  $\Phi, \mathbf{p}$  are calculated as the averages across the last 500 samples.

For BPFA, we have instead chosen to train the  $\beta\gamma\Gamma$ -BPFA model which does infer both  $\mathbf{p}$  and  $\mathbf{r}$  variables, because it is used later on as candidate for comparing different probability models on binarised text. Moreover, we have also considered placing a Gamma prior on the symmetric hyperparameter  $\eta$  of the Dirichlet distributions over  $\Phi$ , which usually provides extra capacity to learn more meaningful topics. The choice of this BPFA model is motivated by its use later in the thesis as a candidate topic model in binarised text. However, we note that the estimators are also valid for BPFA models without prior on  $\eta$  and, obviously, without inference of  $\mathbf{r}$  variables.

The hyperparameters of the  $\beta\gamma\Gamma$ -BPFA are set as follows. We use a Gamma distribution with shape and scale equal to 1 for the prior on the symmetric hyperparameter  $\eta$  of the Dirichlet distribution. The finite Beta Process uses  $\epsilon = 0.005$  and  $c = 1$ . Besides, the Gamma random variables used to mark the Beta Process uses  $c_0 = 1$  and  $r_0 = 1$ . Then, the training is also performed through a Gibbs Sampling algorithm that samples  $\eta, \mathbf{r}$  and the latent counts  $\mathbf{Y} = \{y_{1:}, \dots, y_{1N}\}$  on top of the variables in the  $\beta\Gamma$ -PFA above. We also run the algorithm for 1,000 iterations and we discard a burn-in period of 500 samples. The algorithm also considers an upper-bound for the number of topics  $K_{max}$  which is set to  $\epsilon^{-1} = 200$ . Averages for the point estimates of the global variables are computed across the last 500 samples.

For both models, we have adapted the MATLAB code<sup>3</sup> to use the hyperparameters and settings above, which are also summarised in Table 7.1.

Regarding the samplers, we vary the number of samples up to  $S = 10,000$  for all three methods in both PFA and BPFA, and we use  $S' = 1$  for L2R to keep the same overall number of samples for all estimators.

---

<sup>3</sup>[https://mingyuanzhou.github.io/Softwares/NBP\\_PFA\\_v1.zip](https://mingyuanzhou.github.io/Softwares/NBP_PFA_v1.zip)

| Hyperparameter | $\beta\Gamma$ -PFA | $\beta\gamma\Gamma$ -BPFA                |
|----------------|--------------------|--|
| $\eta$         | 0.1                | $\sim \text{Ga}(1, 1)$                   |
| $K_{max}$      | 100                | 200                                      |
| $\epsilon$     | 0.01               | 0.05                                     |
| $c$            | 1                  | 1  |
| $r_k$          | 1                  | $\sim \text{Ga}(c_0 r_0 = 1, 1/c_0 = 1)$ |
| Burn-in        | 500                | 500                                      |
| Collect        | 500                | 500                                      |

Table 7.1: Hyperparameters and configuration.

### 7.3.2 Experiments with PFA in Downsized Collections

As described in Section 6.3.3, the collections in this experiment are downsized to a vocabulary of 100 words and only documents whose counts give rise to a cardinality of the set  $\mathbb{X}_{y_n}$  smaller than  $10^9$  are kept. As a result, 1,000 documents are considered for each collection from Table 6.2, except for NIPS and AP which only contain 1 and 460 documents with a tractable marginal, respectively.

In this experiment, we also include the L2R with the exact conditionals given by Eqs. (7.8)-(7.10) to compare against the proposed importance sampling approach. Moreover, each experiment is repeated 10 times and we plot their mean and standard error. Fig. 7.1 plots the KL (Kullback-Leibler) divergence between the exact and estimated probabilities as a function of the number of samples for the 4 estimation methods.

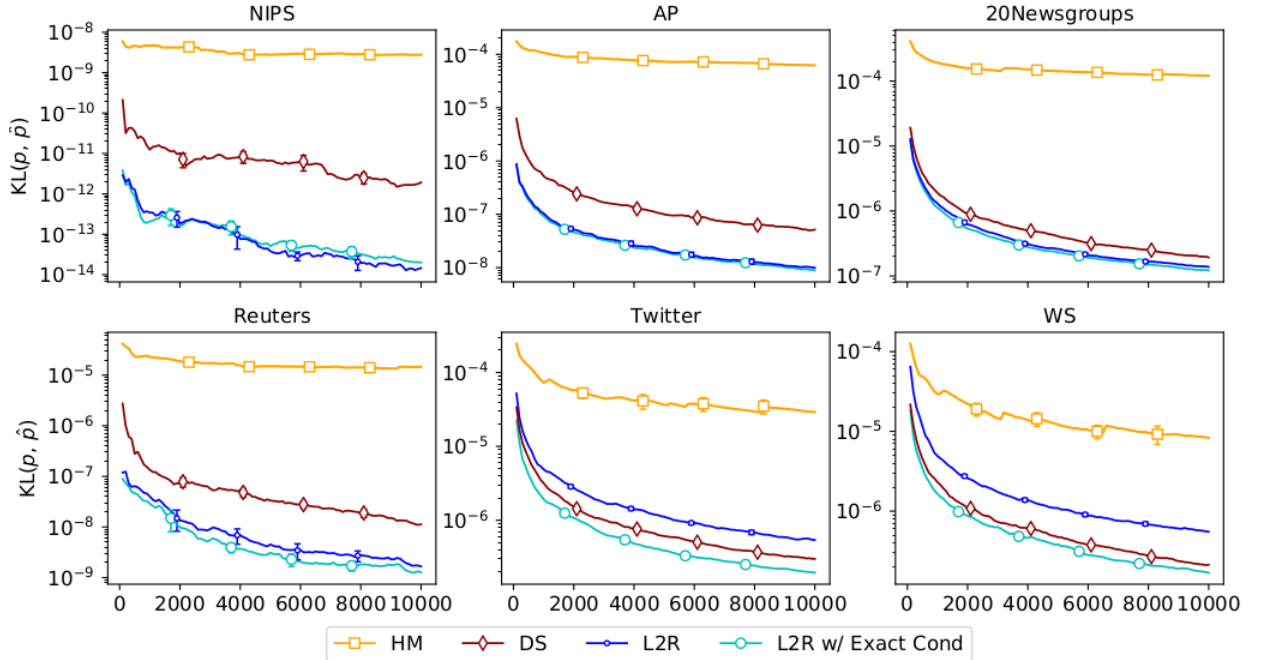


Figure 7.1: Relative Entropy or KL between the estimated document probabilities and the exacts as a function of samples used (Lower KL is better).

Results show that L2R with exact conditionals achieves the lowest KL across all 6 data

sets, followed very closely by the proposed L2R algorithm with  $S' = 1$  which obtains the second lowest KL in 4 data sets. We note that L2R performs worse than DS in Twitter and WS data sets, which both are the shortest text data sets. This poor performance in short text could be explained from the fact that vanilla topic models struggle to learn predictive topic structure due to few word co-occurrence in a document, and hence the proposal in Eq. (7.11) is not close enough to the target to accurately estimate the conditionals with a single sample.

In Fig. 7.2, we have compared the quality of L2R vs DS, as per the results obtained in the last sample of Fig. 7.1, as a function of the average document length of the downsized corpora. We observe that the KL divergences between the exact and approximate estimates in long text data sets are far smaller in L2R than in DS.

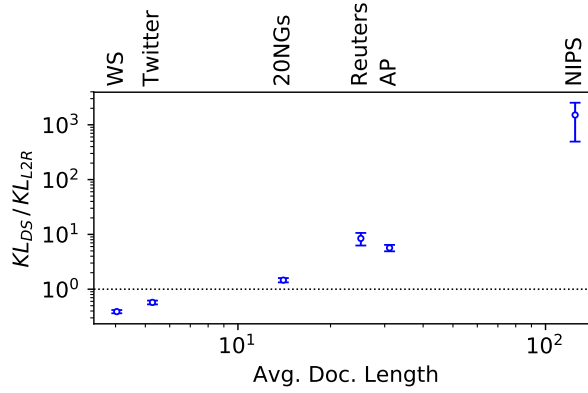


Figure 7.2: KL divergence ratios of L2R and DS estimates.

### 7.3.3 Experiments with PFA in Realistic Collections

In Fig. 7.3, we plot the log-likelihood of 1,000 documents as a function of samples for the three methods that scale to the realistic collections described in Section 6.3.3 and the upper bound for the number of topics set to  $K_{max} = 100$ .

Results show that L2R converges faster than DS in all six collections. The HM method also has a good convergence rate in the four data sets with longest documents, although the inaccuracy reported previously suggests that the method might over-estimate the document likelihood like in LDA (Wallach et al., 2009c; Buntine, 2009). In contrast, the DS method plateaus much slower than any other method across the six collections and specially in long-text, which could indicate that this estimator might be under-estimating the likelihood in high-dimensional scenarios like this.

Therefore, the fast convergence and the fact that its estimates are sandwiched by estimators that tend to under- and over- estimate, validates L2R’s use for document likelihood estimation in PFA with just a few hundred samples.

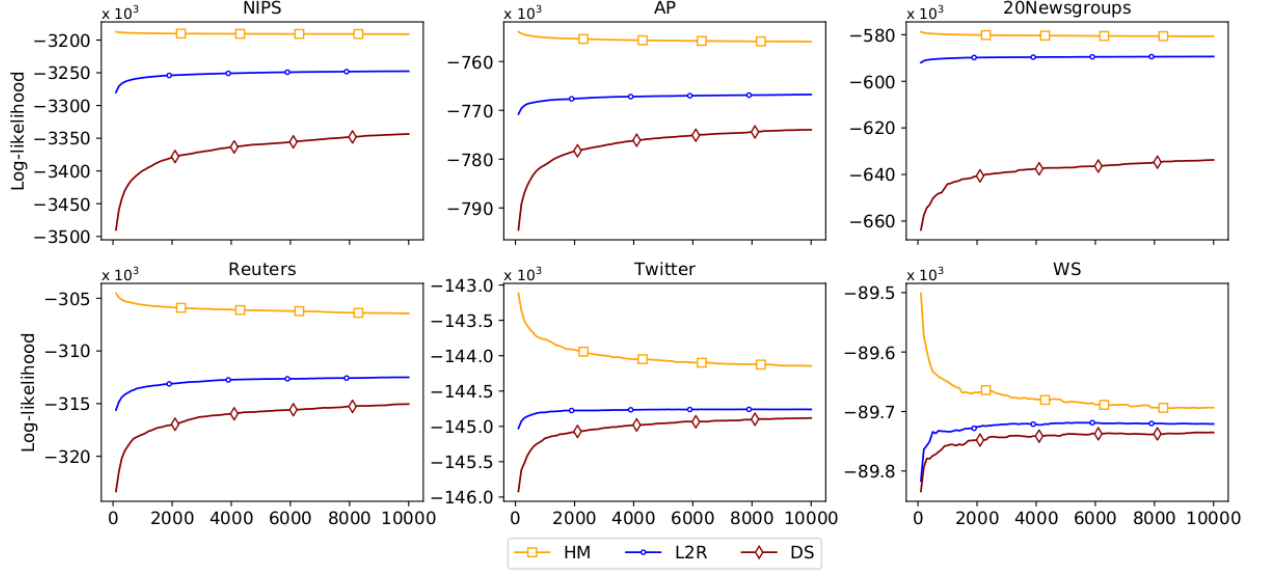


Figure 7.3: Document log-likelihood as a function of the number of samples in PFA.

### 7.3.4 Experiments with BPFA in Realistic Collections

Finally, we present the experiments with BPFA which are conducted on four of the binarised collections in Table 6.2, two representatives of long text (i.e. NIPS and 20Newsgroups) and two of short text (i.e. Twitter, WS).

As shown in Fig. 7.4, the proposed L2R does not converge even with 10,000 samples in long text collections and although its convergence improves in shorter text, it is only comparable to that of the basic DS method. We attribute this poor performance specially in long text to the large sampling space of the left-hand topics described in Section 7.2.1.

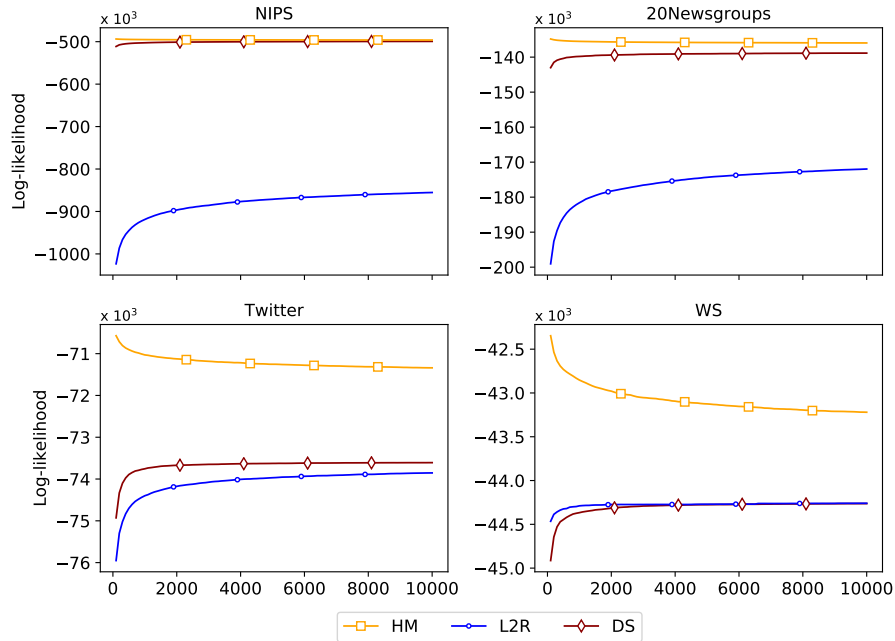


Figure 7.4: Document log-likelihood as a function of the number of samples in BPFA.

## 7.4 Summary and Conclusion

In this chapter, we proposed L2R, a left-to-right sequential sampler for estimating the marginal document likelihood in PFA and BPFA. For both models, we introduced a Gibbs sampling scheme to draw the left-hand topics from the word counts  $y_{n\cdot}$  in PFA and word indicators  $b_{n\cdot}$  in BPFA. For PFA, we showed that the conditional probabilities to the left-hand samples were intractable and hence, we proposed an approximating method based on IS (Importance Sampling). On the contrary, these same conditionals for BPFA were tractable to compute in an exact manner. For both, the estimators could benefit from grouping all zero words at the right-hand side to assess their conditional probability at once and reduce the overall computational cost. Nonetheless, we showed that the time complexity of both estimators is quadratic with the average number of non-zero words.

We then evaluated the L2R samplers in long and short text collections. We analysed the accuracy of the proposed L2R sampler for PFA by comparing its estimates to the exact marginal probability in reasonably small setups. We observed that while it performed similar to the same left-to-right method with exact conditionals in long text, its accuracy dropped in short text because of the approximated conditionals. In more realistic setups, the convergence of the L2R estimator with approximated conditionals by means of one importance sample was faster than any other in both short and long text. These results encourage the use of the L2R sequential sampler for evaluating and comparing PFA topic models, specially in long text collections.

Regarding the convergence analysis of the L2R sampler for BPFA, we showed that the method does not plateau even with 10,000 samples in long text collections. Since this method can compute the conditional probabilities exactly, we associate the poor performance in the binarised model to the unbounded sampling space on the left-hand topics. This hypothesis is reinforced by the fact that the same method in shorter documents, and hence, smaller sampling spaces achieves a very good performance.

In the next chapter, we will explore other estimation methods capable of reducing the time complexity of L2R, which is quadratic in the number of non-zero words, and sometimes impractical for large collections. Moreover, the poor performance of the L2R for BPFA in long text collections also demands to revisit other estimation methods whose sampling process is more focused towards the high probability regions or methods with a completely different sampling space.

# 8

## Mean-field Variational Importance Sampling

*“All exact science is dominated by the idea of approximation”*

BERTRAND Russell

The left-to-right algorithm presented in the previous chapter had a computational cost quadratic with respect to the number of non-zero words, due to the sampling of the topics on the left-hand side of each word in the document. Besides, the performance of this algorithm for binarised counts was specially poor in long text documents due to the unbounded sampling space of topics and words. Therefore, there is a need for more accurate PFA likelihood estimation methods with sub-quadratic cost.

In this chapter, we address the issues above through an alternative algorithm that approximates the marginal document likelihood by means of IS (Importance Sampling)<sup>1</sup> for both PFA (Poisson Factor Analysis) and BPFA (Bernoulli PFA). Instead of leveraging on the chain rule of probability, we leverage on variational inference to define a normalised proposal that approximates the posterior distribution, that is VIS (Variational Importance Sampling). In particular, we consider a variational distribution in the mean-field family, i.e. a fully factorised distribution on the parameters, which we optimise to be close to the posterior in terms of the KL (Kullback-Leibler) divergence. Mean-field importance sampling with the reverse KL divergence was firstly introduced by [Buntine \(2009\)](#) for LDA (Latent Dirichlet Allocation) to frame the IS-IP (Iterated Pseudo-Counts) from [Wallach et al. \(2009c\)](#). However, the use of mean-field VIS in a broader sense to build IS proposal distributions which are tight to the optimal proposal has not yet been explored. We

---

<sup>1</sup>see Section 2.5.3

show that two types of solutions arise from applying the KL divergence in the reverse or forward modes. In particular, we show that the forward KL leads to approximating distributions that might be more suitable for VIS. As a by-product of the KL minimisation, we derive upper and lower bounds to the marginal document likelihood. These bounds are also useful to sandwich the estimates of different methods in realistic scenarios and hence, to determine their accuracy.

We explain in detail the derivation of the mean-field proposals for PFA and BPFA models, which takes advantage of the conditional conjugacy of these models. We show that the lower-bounded proposals can be found through a coordinate ascent algorithm that updates the variational parameters one at a time and the lower bound or ELBO (Evidence Lower Bound) can be assessed in finite time with a tractable closed-form expression. In contrast, the variational parameters for the upper-bounded proposals require of a stochastic algorithm that samples the posterior distribution to approximate the expectations of several sufficient statistics. These samples are also used to approximate the upper bound. Therefore, we also present a Gibbs sampling algorithm to draw these samples from the posterior distribution for both PFA and BPFA.

In the rest of this chapter, we first present the main concepts of mean-field VIS for general LVM (Latent Variable Model) in Section 8.1. In Section 8.2, we derive the lower-bounded and upper-bounded proposals for PFA and their respective optimization algorithms. In Section 8.3, we do the same for BPFA. Then, we evaluate the accuracy and convergence of the proposed VIS methods both in small and realistic scenarios in Section 8.4. Finally, we summarise the main ideas and conclusions of this chapter in Section 8.5.

## 8.1 Mean-Field Variational Importance Sampling

In Section 2.5.3, we introduced the idea behind IS, which we specify next for approximating the marginal likelihood in LVMs. Given a proposal distribution  $Q(z; \gamma)$ , IS approximates the expectation of  $p(x|z; \phi)$  w.r.t.  $p(z)$  by drawing  $S$  samples from the proposal  $z^{(s)} \sim Q(z; \gamma)$  and then, evaluating the weighted average across the likelihood samples  $p(x|z^{(s)}; \phi)$ . That is to say,

$$\begin{aligned} p(x; \phi) &= \int p(x|z; \phi)p(z)dz = \mathbb{E}_{p(z)}p(x|z; \phi) \\ &\approx \frac{1}{S} \sum_{s=1}^S p(x|z^{(s)}; \phi)w(z^{(s)}) \quad \text{where } z^{(s)} \sim Q(z; \gamma) \end{aligned} \quad (8.1)$$

where the weights  $w(z^{(s)})$  are defined as the likelihood ratio  $\frac{p(z^{(s)})}{Q(z^{(s)}; \gamma)}$ . This estimator is unbiased as long as the proposal distribution has the same support as the prior  $p(z)$ . Therefore, the problem of IS boils down to finding a proposal distribution that reduces the variance of this estimator. The optimal proposal  $Q(z; \gamma)$ , in terms of the least variance, is known to be proportional to  $p(x|z; \phi)p(z)$ , but its normalisation constant  $\int p(x|z; \phi)p(z)dz$  is, in fact, the integral that we want to estimate. Thus, the optimal proposal has little practical use despite of the fact that a distribution  $Q(z; \gamma)$ , that is close to the optimal, also has little variance.

A well-known family of distributions used to approximate the posterior in variational inference is the mean-field family. The mean-field family, presented in Section 2.5.2, assumes



a fully factorised distribution across variables and each distribution is governed by its own variational parameters. That is, we introduce mean-field VIS by specifying the proposal  $Q(z; \gamma)$  in Eq. (8.1) to be in the form of,

$$Q(z; \gamma) = \prod_{k=1}^K Q_k(z_k; \gamma_k) \quad (8.2)$$

where the random variables  $z_k$  are assumed to be independent, distributed according to  $Q_k(\cdot)$  and governed by its own variational parameters  $\gamma_k$ . Mean-field importance sampling was first mentioned in (Buntine, 2009), who showed that the IS-IP method from Wallach et al. (2009c) could be understood in these terms.

We then impose that the mean-field distribution approximates the optimal proposal in terms of KL divergence. The KL is an asymmetric divergence that leads to different types of solutions depending on the order of its elements. Whereas variational inference has classically considered the reverse KL because its optimization leads to simple coordinate ascent algorithms, we also considered here the forward KL. The forward KL divergence between an optimal proposal distribution <sup>2</sup>  $p(z|x; \phi)$  and the approximating distribution  $Q_U(z; \gamma_U)$  is defined as,

$$\text{KL}(p(z|x; \phi), Q_U(z; \gamma_U)) = \int p(z|x; \phi) \log \frac{p(z|x; \phi)}{Q_U(z; \gamma_U)} dz = \mathbb{E}_{p(z|x; \phi)} \log \frac{p(z|x; \phi)}{Q_U(z; \gamma_U)} \quad (8.3)$$

where  $\mathbb{E}_{p(z|x; \phi)}$  refers to the expectation w.r.t. the posterior  $p(z|x; \phi)$ . And the reverse KL between the approximating distribution  $Q_L(z; \gamma_L)$  and an optimal proposal  $p(z|x; \phi)$  is given by,

$$\text{KL}(Q_L(z; \gamma_L), p(z|x; \phi)) = \int Q_L(z; \gamma_L) \log \frac{Q_L(z; \gamma_L)}{p(z|x; \phi)} dz = \mathbb{E}_{Q_L(z)} \log \frac{Q_L(z; \gamma_L)}{p(z|x; \phi)} \quad (8.4)$$

where  $\mathbb{E}_{Q_L(z)}$  is the expectation w.r.t. the proposal  $Q_L(z; \gamma_L)$ .

The difference between the type of approximations created by the reverse and forward KL are depicted in Fig. 8.1. Whereas the reverse KL leads to solutions  $Q_L(z; \gamma_L)$  that force zeros wherever  $p(z|x; \phi)$  is zero, the solutions for the forward KL  $Q_U(z; \gamma_U)$  can be non-zeros in these regions, as Eq. (8.3) (8.4) suggest. This causes the reverse KL to focus on a single mode whereas the forward KL can span across several modes. As a result, we expect the proposal  $Q_U(z; \gamma_U)$  to be more suitable for approximating multi-modal distributions, since it will enable the different modes to be sampled  $z^{(s)} \sim Q_U(z; \gamma_U)$ .

However, the direct optimization of the KL divergences is not trivial because both contain unknown posterior probabilities  $p(z|x)$ . One approach to optimise the KL divergence consists in building a surrogate objective by bounding the marginal document likelihood (Ji et al., 2010). In the next section, we explain how to build a lower and upper bound to the marginal document likelihood and how these bounds are related to the KL divergences, presented above.

---

<sup>2</sup>see the definition of optimal  $Q(z) \propto |f(z)|p(z)$  for  $f(z) = p(x|z; \phi)$  and  $p(z) = p(z)$  in Section 23.4 from Murphy (2012)

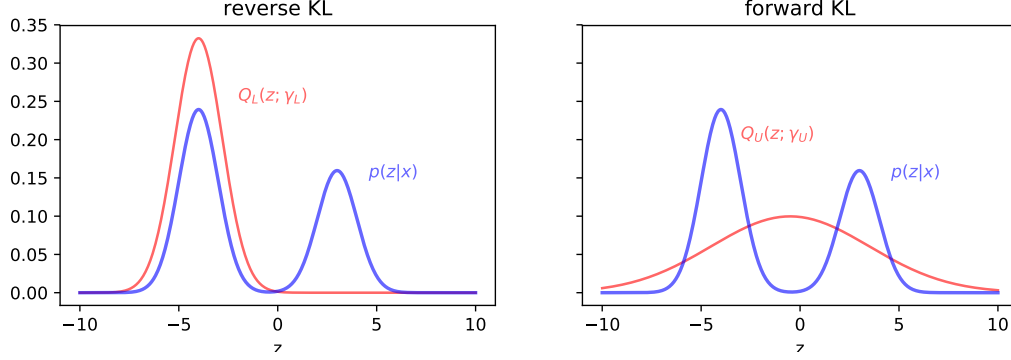


Figure 8.1: Reverse vs. forward KL divergence for a mixture of univariate Gaussians (blue) approximated by a Gaussian distribution (red).

### 8.1.1 Variational Lower Bounds

To build a lower bound to the logarithm of the marginal likelihood, we first multiply and divide the likelihood  $p(x|z; \phi)$  inside the expectation by the variational distribution  $Q_L(z; \gamma^L)$ ,

$$\log p(x; \phi) = \log \mathbb{E}_{p(z)}[p(x|z; \phi)] = \log \mathbb{E}_{p(z)} \left[ p(x|z; \phi) \frac{Q_L(z; \gamma^L)}{Q_L(z; \gamma^L)} \right] \quad (8.5)$$

and then, we swap the expectation w.r.t.  $p(z)$  for that w.r.t.  $Q_L(z; \gamma^L)$

$$\log p(x; \phi) = \log \mathbb{E}_{Q_L(z; \gamma^L)} \left[ \frac{p(x|z; \phi)p(z)}{Q_L(z; \gamma^L)} \right] = \log \mathbb{E}_{Q_L(z; \gamma^L)} \left[ \frac{p(x, z; \phi)}{Q_L(z; \gamma^L)} \right] \quad (8.6)$$

where we use  $p(x, z; \phi) = p(x|z; \phi)p(z)$ .

Finally, we build the lower bound to the marginal likelihood by applying the Jensen's Inequality to,

$$\log p(x; \phi) = \log \mathbb{E}_{Q_L(z; \gamma^L)} \left[ \frac{p(x, z; \phi)}{Q_L(z; \gamma^L)} \right] \geq \mathbb{E}_{Q_L(z; \gamma^L)} \left[ \log \frac{p(x, z; \phi)}{Q_L(z; \gamma^L)} \right] = U_L \quad (8.7)$$

where the logarithm can be pushed inside of the expectation because the log is a convex downward function.

This bound, also known as the ELBO in variational inference, is tight when the approximating distribution  $Q_L(z; \gamma^L)$  is close to the posterior distribution  $p(z|x)$  in terms of reverse KL divergence, as expressed by,

$$U_L = \log p(x; \phi) - \text{KL}(Q_L(z; \gamma^L), p(z|x; \phi)). \quad (8.8)$$

which can be derived from Eq. (8.7) by expressing the joint distribution as  $p(x, z; \phi) = p(x; \phi)p(z|x; \phi)$  and applying basic properties of logarithms to make the KL in Eq. (8.4) appear.

Therefore, we can minimise the reverse KL divergence by instead maximising the lower bound in Eq. 8.7. The fact that this lower bound does not depend on the posterior  $p(z|x)$  allows us to optimise for the variational distribution  $Q_L(z; \gamma^L)$ . When this variational distribution is chosen to be in the mean-field family as in Eq. (8.2), this optimization can be done independently for each mean-field distribution. Furthermore, analytical mean-field distributions can be derived for conditionally conjugate models, as we will show later for the augmented PFA and BPFA models.

### 8.1.2 Variational Upper Bound

To build an upper bound to the marginal likelihood, we first add and subtract the logarithm of the posterior distribution  $\log p(z|x; \phi)$  to the marginal document log-likelihood,

$$\log p(x; \phi) = \log p(x, z; \phi) + \log p(z|x; \phi) - \log p(z|x; \phi) \quad (8.9)$$

where the first two terms can be combined into the logarithm of their joint distribution as shown next,

$$\log p(x; \phi) = \log p(x, z; \phi) - \log p(z|x; \phi). \quad (8.10)$$

Given that the marginal document likelihood does not depend on  $z$ , we can then take expectations w.r.t the posterior on  $z$  on the right-hand side of the equality,

$$\begin{aligned} \log p(x; \phi) &= \mathbb{E}_{p(z|x; \phi)} [\log p(x, z; \phi) - \log p(z|x; \phi)] \\ &= \mathbb{E}_{p(z|x; \phi)} [\log p(x, z; \phi)] + \mathbb{H}(z|x; \phi) \end{aligned} \quad (8.11)$$

where  $\mathbb{H}(z|x; \phi)$  is the entropy of the random variable  $z$  conditioned on  $x$ .

Finally, we apply the Gibbs' inequality to upper bound the marginal document likelihood in the previous equation as follows,

$$\log p(x; \phi) \leq \mathbb{E}_{p(z|x; \phi)} [\log p(x, z; \phi)] - \mathbb{E}_{p(z|x; \phi)} [\log Q_U(z; \gamma^U)] \quad (8.12)$$

which says that the entropy of the posterior on  $z$  is less or equal to its cross-entropy with any other distribution  $Q_U(z; \gamma^U)$ . We can rearrange the logarithmic terms in the equation above to present the upper bound as,

$$\log p(x; \phi) \leq \mathbb{E}_{p(z|x; \phi)} \left[ \log \frac{p(x, z; \phi)}{Q_U(z; \gamma^U)} \right] = U_U. \quad (8.13)$$

This bound, also known as EUBO (Evidence Upper BOUND), is tight when  $Q_U(z; \gamma^U)$  is close to the posterior  $p(z|x; \phi)$  in terms of KL divergence, expressed as,

$$U_U = \log p(x; \phi) + \text{KL}(p(z|x; \phi), Q_U(z; \gamma^U)). \quad (8.14)$$

which can be derived from Eq. (8.13) by expressing the joint distribution as  $p(x, z; \phi) = p(x; \phi)p(z|x; \phi)$  and applying basic properties logarithms to make the KL in Eq. (8.3) appear.

Therefore, minimising the forward KL divergence is equivalent to minimise their upper bounds  $U_U$  w.r.t the variational distributions  $Q_U(z; \gamma^U)$ . However, the upper bound in Eq. (8.13) still involves the posterior expectations. Nonetheless, when the variational distributions are chosen to be in the mean-field family and specific forms are imposed to the mean-field distributions, we can still optimise this objective w.r.t the variational parameters  $\gamma^U$ , as we show later for PFA and BPFA.

## 8.2 Mean-field VIS for PFA

We define the VIS estimator for PFA through the augmented model in Section 2.4.6, because this model allows us to marginalize the continuous factors  $l_{n:}$  as in Eq. (6.4) and hence, to simply sample the discrete topic counts  $x_{n:}$ . It is known that the sampling of continuous

values may lead to estimators with infinite variance, so it is a good practice to collapse first continuous variables when possible (Wallach et al., 2009c). Thus, the marginal document likelihood for PFA can be expressed as,

$$p(y_{n:}; \Phi, \mathbf{p}, \mathbf{r}) = \sum_{x_{n::} \in \mathbb{X}_{y_{n:}}} p(x_{n::}; \Phi, \mathbf{p}, \mathbf{r}) \quad (8.15)$$

where the sum is over all matrices of non-negative integers whose rows add up to the observed counts  $\mathbb{X}_{y_{n:}} = \{x_{n::} \in \mathbb{N}_0^{V \times K} \mid y_{n:} = \sum_{k=1}^K x_{n:k}\}$  and the summands are computed as,

$$p(x_{n::}; \Phi, \mathbf{p}, \mathbf{r}) = \prod_{k=1}^K \text{NM} \left( x_{n:k}; r_k, \frac{p_k \phi_k}{1 - p_k + p_k \sum_p \phi_{kp}} \right) \quad (8.16)$$

the product of  $K$  NMs (Negative Multinomials). To avoid the sum over this constrained set, we introduce a proposal distribution  $Q_{y_{n:}}(x_{n::} | \gamma_n)$ , whose support is in  $\mathbb{X}_{y_{n:}}$  and is parametrised with the variational parameter  $\gamma_n$ . By multiplying the numerator and denominator with the proposal distribution, we can express the marginal likelihood as,

$$p(y_{n:}; \Phi, \mathbf{p}, \mathbf{r}) = \sum_{x_{n::} \in \mathbb{X}_{y_{n:}}} \frac{p(x_{n::}; \Phi, \mathbf{p}, \mathbf{r})}{Q_{y_{n:}}(x_{n::}; \gamma_n)} Q_{y_{n:}}(x_{n::}; \gamma_n) = \mathbb{E}_{Q_{y_{n:}}(x_{n::}; \gamma_n)} \frac{p(x_{n::}; \Phi, \mathbf{p}, \mathbf{r})}{Q_{y_{n:}}(x_{n::}; \gamma_n)} \quad (8.17)$$

where  $\mathbb{E}_{Q_{y_{n:}}(x_{n::}; \gamma_n)}$  is the expectation w.r.t. the proposal distribution. Finally, one can approximate this expectation via sampling by simply,

$$p(y_{n:}; \Phi, \mathbf{p}, \mathbf{r}) \approx \frac{1}{S} \sum_{s=1}^S \frac{p(x_{n::}^{(s)}; \Phi, \mathbf{p}, \mathbf{r})}{Q_{y_{n:}}(x_{n::}^{(s)}; \gamma_n)} \quad \text{where } x_{n::}^{(s)} \sim Q_{y_{n:}}(x_{n::}; \gamma_n) \quad (8.18)$$

where  $x_{n::}^{(s)}$  are the samples drawn from the conditioned proposal distribution. Note that  $Q_{y_{n:}}(x_{n::}; \gamma_n)$  refers to both proposals  $Q_{U_{y_{n:}}}(x_{n::}; \gamma_n^U)$  and  $Q_{L_{y_{n:}}}(x_{n::}; \gamma_n^L)$  build either from minimising the forward KL or the backward KL and their support is in  $\mathbb{X}_{y_{n:}}$ . For notation clarity, we omit the subscript  $y_{n:}$  in the proposal distribution from now onwards. Furthermore, we note that the importance weights in Eq. (8.18) are not as explicitly stated as in Eq. (8.1), because the dependency between  $x_{n::}$  and  $y_{n:}$  is deterministic.

Despite the benefit of sampling from a discrete distribution, the distribution  $p(x_{n::}; \Phi, \mathbf{p}, \mathbf{r})$  does not enable to directly obtain analytic variational distributions across words because its complete conditionals are not in the exponential family and the closed-form mean-field distributions cannot be derived from them. To circumvent this, we can learn a mean-field distribution for an augmented model, which might include other variables beyond  $x_{n::}$ , and then, use as proposal the mean-field distributions associated to  $x_{n::}$  to perform VIS in Eq. (8.18).

In the following sections, we derive the mean-field distributions for the augmented PFA model in Section 2.4.6, in which the factor  $l_{n:}$  are not collapsed, and then we only use as proposal the mean-field distributions associated to  $x_{n::}$ .

### 8.2.1 Lower-bounded Mean-field Proposal

In the next two sections we first show how to maximise the lower bound from Section 8.1.1 for the PFA and then, how to compute its value.

### 8.2.1.1 Maximising the Lower Bound

The ELBO in Eq. 8.7 can be rewritten for the augmented PFA model as,

$$\begin{aligned} U_L &= \mathbb{E}_{Q_{L_{y_{n:}}}(x_{n:}, l_{n:}; \gamma_n^L, \beta_n^L)} \log \frac{p(y_{n:}, x_{n:}, l_{n:}; \Phi, \mathbf{p}, \mathbf{r})}{Q_{L_{y_{n:}}}(x_{n:}, l_{n:}; \gamma_n^L, \beta_n^L)} \\ &= \mathbb{E}_{Q_{L_{y_{n:}}}(x_{n:}, l_{n:}; \gamma_n^L, \beta_n^L)} \log p(y_{n:}, x_{n:}, l_{n:}; \Phi, \mathbf{p}, \mathbf{r}) + \mathbb{H}_{y_{n:}}(x_{n:}, l_{n:}; \gamma_n^L, \beta_n^L) \end{aligned} \quad (8.19)$$

where the joint distribution of the observed word counts  $y_{n:}$ , the latent topic counts  $x_{n:}$  and factors  $l_{n:}$  is given by,

$$p(y_{n:}, x_{n:}, l_{n:}; \Phi, \mathbf{p}, \mathbf{r}) = \prod_{k=1}^K \prod_{p=1}^V \text{Pois}(x_{npk} | l_{nk} \phi_{kp}) \text{Ga} \left( l_{nk}; r_k, \frac{p_k}{1 - p_k} \right) \quad (8.20)$$

with  $x_{n:} \in \mathbb{X}_{y_{n:}}$ . Besides, the lower-bounded proposal  $Q_{L_{y_{n:}}}(x_{n:}, l_{n:}; \gamma_n^L, \beta_n^L)$  is the mean-field distribution that factorises across words  $V$  and factors  $K$  as follows,

$$Q_{L_{y_{n:}}}(x_{n:}, l_{n:}; \gamma_n^L, \beta_n^L) = \prod_{p=1}^V Q_{L_{y_{np}}}(x_{np:}; \gamma_{np}^L) \prod_{k=1}^K Q_{L_{y_{np}}}(l_{nk}; \beta_{nk}^L) \quad (8.21)$$

Moreover, the augmented PFA model is conditionally conjugate and its complete conditionals can be written in the exponential family as,

$$p(x_{np:} | y_{n:}, x_{n-\neg p}, l_{n:}, \Phi, \mathbf{p}, \mathbf{r}) = \text{Mult}(x_{np:} | y_{np}, \frac{l_{n:} \phi_{:p}^T}{\sum_{k=1}^K l_{nk} \phi_{kp}^T}) \quad (8.22)$$

$$p(l_{nk} | y_{n:}, l_{n-\neg k}, x_{n:}, \Phi, \mathbf{p}, \mathbf{r}) = \text{Ga}(l_{nk} | r_k + \sum_{p=1}^V x_{npk}, p_k) \quad (8.23)$$

where  $\text{Mult}(\cdot)$  and  $\text{Ga}(\cdot)$  are Multinomial and Gamma distributions parametrised as in Eq. (B.4) and Eq. (B.9), respectively. Under these circumstances, the mean-field distributions that optimises the lower bound in Eq. (8.19) has the same analytic expression than the corresponding complete conditionals

$$\begin{aligned} Q_{L_{y_{np}}}(x_{np:} | \gamma_{np}^L) &= \text{Mult}(x_{np:} | y_{np}, \gamma_{np}^L) \\ Q_{L_{y_{np}}}(l_{nk} | \beta_{nk}^L) &= \text{Ga}(l_{nk} | \beta_{nk1}^L, \beta_{nk2}^L) \end{aligned} \quad (8.24)$$

where their variational parameters  $\gamma_{np}^L, \beta_{nk}^L$  can be computed in their natural form as the expectations of the natural parameters of the complete conditionals,

$$\eta(\gamma_{np}^L) = \mathbb{E}_{Q_{L_{y_{np}}}(l_{n:} | \beta_n^L)} [\eta(x_{np:})] = \begin{bmatrix} \mathbb{E}_{Q_{L_{y_{np}}}(l_{n1} | \beta_n^L)} [\log l_{n1}] + \log \phi_{1p} + C \\ \vdots \\ \mathbb{E}_{Q_{L_{y_{np}}}(l_{nK} | \beta_n^L)} [\log l_{nK}] + \log \phi_{Kp} + C \end{bmatrix} \quad (8.25)$$

$$\eta(\beta_{nk}^L) = \mathbb{E}_{Q_{L_{y_{np}}}(x_{n:} | \gamma_n^L)} [\eta(l_{nk})] = \begin{bmatrix} r_k + \sum_{p=1}^V \mathbb{E}_{Q_{L_{y_{np}}}(x_{np:} | \gamma_n^L)} [x_{npk}] - 1 \\ -\frac{1}{p_k} \end{bmatrix}. \quad (8.26)$$

Note that the expectations above are taken w.r.t. the mean-field distribution without the factor corresponding to the variable of interest ( $x_{np}$  in Eq. (8.25) and  $l_{nk}$  in Eq. (8.26)). These expectations can be calculated analytically by exploiting the property of the exponential family that the derivatives of the cumulant w.r.t the natural parameters correspond to the expectations of the sufficient statistics. Therefore, the expected logarithm of a gamma random variable is given by,

$$\mathbb{E}_{Q_{Ly_{np}}(l_{nk}|\beta_n^L)} [\log l_{nk}] = \frac{\delta A(\eta(\beta_{nk}^L))}{\delta \beta_{nk1}^L} = \Psi(\beta_{nk1}^L) + \log \beta_{nk2}^L \quad (8.27)$$

where  $\Psi(\cdot)$  is the digamma function. Besides, the expectation of a Multinomial random variable is simply its mean value,

$$\mathbb{E}_{Q_{Ly_{np}}(x_{np}|\gamma_n^L)} [x_{npk}] = y_{np} \gamma_{npk}^L. \quad (8.28)$$

Finally, the variational parameters can be written in the original parametrization of the Multinomial and Gamma distributions as follows,

$$\xi_{np:}^L \leftarrow \phi_{:w} \beta_{n:2}^L e^{\Psi(\beta_{n:1}^L)} \quad (8.29)$$

$$\gamma_{np:}^L \leftarrow \frac{\xi_{np:}^L}{\sum_{k=1}^K \xi_{npk}^L} \quad (8.30)$$

$$\beta_{nk1}^L \leftarrow r_k + \sum_{p=1}^V y_{np} \gamma_{npk}^L \quad \beta_{nk2}^L \leftarrow p_k \quad (8.31)$$

where the dependency between the updates of  $\xi_{np:}^L$ ,  $\gamma_{np:}^L$  and  $\beta_{nk1}^L$  forces us into a coordinate ascent algorithm.

### 8.2.1.2 Computing the Lower Bound

To assess the convergence of the coordinate ascent, we can use the same lower bound in Eq. (8.19) for which we can derive a closed-form expression as follows.

Thanks to the mean-field factorisation, the entropy term in Eq. (8.19) can be calculated by summing the entropy associated to each random variable,

$$\mathbb{H}_{y_n.}(x_{n:}, l_{n:}; \gamma_n^L, \beta_n^L) = \sum_{p=1}^V \mathbb{H}_{y_n.}(x_{np:}; \gamma_{np:}^L) + \sum_{k=1}^K \mathbb{H}_{y_n.}(l_{nk} | \beta_{nk1}^L, \beta_{nk2}^L) \quad (8.32)$$

where the entropy for a Multinomial random variable can be found in Appendix E.1 and that of a Gamma random variable in Appendix E.2. Besides, the expectation term in Eq. (8.19) requires to take into account the factorisation of the graphical model in order to push the expectations inside. With that, we can calculate the corresponding expectation for each variable individually and sum them as follows,

$$\begin{aligned} \mathbb{E}_{Q_{Ly_n.}(x_{n:}, l_{n:}; \gamma_n^L, \beta_n^L)} \log p(x_{n:}, l_{n:}; \Phi, \mathbf{p}, \mathbf{r}) &= \sum_{k=1}^K \sum_{p=1}^V \mathbb{E}_{Q_{Ly_{np}}(x_{np:}, l_{nk}; \gamma_{np:}^L, \beta_{nk1}^L, \beta_{nk2}^L)} \log p(x_{npk} | l_{nk}, \phi_{kp}) \\ &+ \sum_{k=1}^K \mathbb{E}_{Q_{Ly_{np}}(l_{nk}; \beta_{nk1}^L, \beta_{nk2}^L)} \log p(l_{nk} | p_k, r_k) \end{aligned} \quad (8.33)$$

where the expectation of the logarithm of the Poisson distribution can be found in Appendix E.3 and the expectation of the logarithm of a Gamma distribution in Appendix E.4.

### 8.2.1.3 Coordinate Ascent Algorithm

In Algorithm 8.1, we depict the pseudocode to compute the parameters for the Lower-bounded proposal  $Q_{L_{y_n}}(x_{n::}, l_{n:}; \gamma_n^L, \beta_n^L)$  as well as the lower bound  $U_L$ , based on the derivations above. Given that the VIS estimator defined in Eq. (8.18) only requires the marginal on the topic counts, the algorithm simply returns  $\gamma_{n::}^L$ , but note that it also computes  $\beta_{n::}^L$ . The coordinate ascent part has a computational cost linear in the number of topics  $K$  and the number of non-zero words  $V_{c_n}$ , whereas the function lower bound computation,  $\text{ComputeELBO}()$ , is linear in  $K$ ,  $V_{c_n}$ , but also in the maximum word count in  $y_n$ .

---

**Algorithm 8.1:** PFA Lower-bounded Proposal.

---

```

input :  $y_{n::}, \mathbf{r}, \mathbf{p}, \Phi$ 
output:  $\gamma_{n::}^L$ 

1  $V_{c_n} \leftarrow |y_n: > 0|$ 
2  $K \leftarrow \text{Length}(\mathbf{p})$ 
3 Function  $\text{ComputeELBO}(y_{n::}, \mathbf{r}, \mathbf{p}, \Phi, \gamma_{n::}^L, \beta_{n::}^L)$ 
4    $U^L \leftarrow 0$ 
5   for  $p \leftarrow 1$  to  $V_{c_n}$  do
6      $U^L \leftarrow U^L + \text{EntropyMult}(y_{np}, \gamma_{np:}^L)$  Eq. (E.5)
7   for  $k \leftarrow 1$  to  $K$  do
8      $U^L \leftarrow U^L + \text{EntropyGa}(\beta_{nk:}^L)$  Eq. (E.12)
9      $U^L \leftarrow U^L + \text{ExpLogGa}(r_k, p_k, \beta_{nk:}^L)$  Eq. (E.19)
10    for  $p \leftarrow 1$  to  $V_{c_n}$  do
11       $U^L \leftarrow U^L + \text{ExpLogPois}(y_{np}, \phi_{kp}, \gamma_{npk}^L, \beta_{nk:}^L)$  Eq. (E.17)
12  return  $U^L$ 

  /* Coordinate ascent algorithm */
13 for  $k \leftarrow 1$  to  $K$  do
14    $\beta_{nk:}^L \leftarrow \text{Init}(r_k, p_k)$ 
15 while  $U_L \leftarrow \text{ComputeELBO}(y_{n::}, \mathbf{r}, \mathbf{p}, \Phi, \gamma_{n::}^L, \beta_{n::}^L)$  not converged do
16   for  $p \leftarrow 1$  to  $V_{c_n}$  do
17     for  $k \leftarrow 1$  to  $K$  do
18        $\xi_{npk}^L \leftarrow \text{UpdateXi}(y_{np}, \phi_{kp}, \beta_{nk:}^L)$  Eq. (8.29)
19        $\gamma_{np:}^L \leftarrow \text{UpdateGamma}(\xi_{np:}^L)$  Eq. (8.30)
20   for  $k \leftarrow 1$  to  $K$  do
21      $\beta_{nk:}^L \leftarrow \text{UpdateBeta}(y_{n:}, \gamma_{n:k}^L)$  Eq. (8.31)

```

---

## 8.2.2 Upper-bounded Mean-field Proposal

In the next two sections we first show how to minimise the upper bound from Section 8.1.2 for the PFA and then, how to approximate its value.

### 8.2.2.1 Minimising the Upper Bound

Similar to the ELBO, we can write the EUBO in Eq. (8.13) for the augmented PFA model as,

$$\begin{aligned} U_U &= \mathbb{E}_{p(x_{n::}, l_{n:} | y_{n:}; \Phi, \mathbf{p}, \mathbf{r})} \log \frac{p(y_{n:}, x_{n::}, l_{n:}; \Phi, \mathbf{p}, \mathbf{r})}{Q_{U_{y_{n:}}}(x_{n::}, l_{n:}; \gamma_n^U, \beta_n^U)} \\ &= \mathbb{E}_{p(x_{n::}, l_{n:} | y_{n:}; \Phi, \mathbf{p}, \mathbf{r})} \log p(y_{n:}, x_{n::}, l_{n:}; \Phi, \mathbf{p}, \mathbf{r}) - \mathbb{E}_{p(x_{n::}, l_{n:} | y_{n:}; \Phi, \mathbf{p}, \mathbf{r})} \log Q_{U_{y_{n:}}}(x_{n::}, l_{n:}; \gamma_n^U, \beta_n^U) \end{aligned} \quad (8.34)$$

where the joint distribution on the observed word counts  $y_{n:}$ , the latent topic counts  $x_{n::}$  and factors  $l_{n:}$  is the same as in Eq. (8.20), but the posterior  $p(x_{n::}, l_{n:} | y_{n:}; \Phi, \mathbf{p}, \mathbf{r})$  is intractable to compute. This complex posterior is what hampers the derivation of closed-form updates for this upper-bounded proposal.

One approach to solve this problem is to consider specific forms for the mean-field distributions. In our case, we assume the same statistical forms than the mean-field distributions in the lower bound case:  $l_{nk}$  is distributed according to a Gamma and each  $x_{nw:}$ , follows a Multinomial. That is,

$$Q_{U_{y_{n:}}}(x_{n::}, l_{n:} | \gamma_{n::}^U, \beta_{n::}^U) = \prod_{p=1}^V \text{Mult}(x_{np:} | y_{np}, \gamma_{np:}^U) \prod_{k=1}^K \text{Ga}(l_{nk} | \beta_{nk1}^U, \beta_{nk2}^U). \quad (8.35)$$

Then, we seek to minimise the upper bound in Eq. (8.34) with respect to the variational parameters,  $\gamma_{np:}^U$ ,  $\beta_{nk1}^U$  and  $\beta_{nk2}^U$ . Mathematically speaking, we would like to find the variational parameters that equal the gradients of the upper bound to 0,

$$\nabla_{\gamma_{np:}^U, \beta_{nk1}^U, \beta_{nk2}^U} U_U(\gamma_{np:}^U, \beta_{nk1}^U, \beta_{nk2}^U) = 0, \quad (8.36)$$

where we note that the first expectation in Eq. (8.34) does not depend on the variational parameters and hence, its gradient is 0. As a result of this, we focus next on the second expectation,

$$-\nabla_{\gamma_{np:}^U, \beta_{nk1}^U, \beta_{nk2}^U} \mathbb{E}_{p(x_{n::}, l_{n:} | y_{n:}; \Phi, \mathbf{p}, \mathbf{r})} \log Q_{U_{y_{n:}}}(x_{n::}, l_{n:}; \gamma_n^U, \beta_n^U) = 0 \quad (8.37)$$

Because the expectation is w.r.t. the posterior which does not depend on the variational parameters, the gradient can be pushed inside the expectation,

$$\mathbb{E}_{p(x_{n::}, l_{n:} | y_{n:}; \Phi, \mathbf{p}, \mathbf{r})} \nabla_{\gamma_{np:}^U, \beta_{nk1}^U, \beta_{nk2}^U} \log Q_{U_{y_{n:}}}(x_{n::}, l_{n:}; \gamma_n^U, \beta_n^U) = 0 \quad (8.38)$$

and the upper-bounded proposal, which factors across variables, can be decomposed into a sum of logarithmic distributions and each variational parameter solved independently. Furthermore, the fact that each mean-field distribution is in the exponential family implies that the logarithm of these distributions will be a convex function w.r.t. its parameters. Therefore, the global minimum can be found by solving the following system of equations,

$$\mathbb{E}_{p(x_{n::}, l_{n:} | y_{n:}; \Phi, \mathbf{p}, \mathbf{r})} [t(x_{np:})] \frac{\delta \eta(\gamma_{np:})}{\delta \gamma_{np:}} - \frac{\delta A(\eta(\gamma_{np:}))}{\delta \gamma_{np:}} = 0 \quad p = 1 \dots V \quad (8.39)$$

$$\mathbb{E}_{p(x_{n::}, l_{n:} | y_{n:}; \Phi, \mathbf{p}, \mathbf{r})} [t(l_{nk})] \frac{\delta \eta(\beta_{nk:})}{\delta \beta_{nk:}} - \frac{\delta A(\eta(\beta_{nk:}))}{\delta \beta_{nk:}} = 0 \quad k = 1 \dots K \quad (8.40)$$



where the mean-field distributions have been expressed in the exponential family to show the structure of the problem. Eq. (8.39) is the equation that results from taking the partial derivative in Eq. (8.38) w.r.t the variational parameters associated with variable  $x_{np:}$ , which is assumed to be Multinomial. Thus,  $t(x_{np:})$  is the vector of sufficient statistics,  $\eta(\gamma_{np:})$  is the vector of natural parameters and  $A(\eta(\gamma_{np:}))$ , the cumulant displayed in Table B.1 for the Multinomial distribution. Similarly, Eq. (8.40) is the equation that refers to the Gamma random variable  $l_{nk}$  with sufficient statistics  $t(l_{nk})$ , natural parameters  $\eta(\beta_{nk:})$  and cumulant  $A(\eta(\beta_{nk:}))$ . In summary, each equation in the system can be derived as the expectation w.r.t. the posterior distribution of the sufficient statistics of that variable times the partial derivative of the natural parameters w.r.t. the original parametrization minus the partial derivatives of the cumulant.

In particular, the system for PFA is described by the following equations,

$$\begin{bmatrix} \mathbb{E}_p[x_{np1}] \\ \vdots \\ \mathbb{E}_p[x_{npK}] \end{bmatrix} \begin{bmatrix} \frac{1}{\gamma_{np1}} \\ \vdots \\ \frac{1}{\gamma_{npK}} \end{bmatrix} = \begin{bmatrix} \lambda \\ \vdots \\ \lambda \end{bmatrix} \quad p = 1 \dots V \quad (8.41)$$

$$\begin{bmatrix} \mathbb{E}_p[\log l_{nk}] \\ \mathbb{E}_p[l_{nk}] \end{bmatrix} \begin{bmatrix} 1 \\ 1/\beta_{nk2}^U \end{bmatrix} - \begin{bmatrix} \Psi(\beta_{nk1}^U) + \log \beta_{nk2}^U \\ \beta_{nk1}^U / \beta_{nk2}^U \end{bmatrix} = \begin{bmatrix} 0 \\ 0 \end{bmatrix} \quad k = 1 \dots K \quad (8.42)$$

where  $\lambda$  is the Lagrange multiplier that accounts for the restriction  $\sum_{k=1}^K \gamma_{npk} = 1$  and  $\mathbb{E}_p$  refers to expectation w.r.t the posterior  $p(x_{n::}, l_{n:} | y_{n:}; \Phi, \mathbf{p}, \mathbf{r})$ . While this system can be solved analytically for  $\gamma_{np:}$ , the non-linearity in the  $\beta_{nk:}$  parameters requires to solve them numerically. We propose to use the Newton's method to iteratively solve Eq. (8.45) below,

$$\xi_{np:}^U = \mathbb{E}_p[x_{np:}] \quad (8.43)$$

$$\gamma_{np:}^U = \frac{\xi_{np:}^U}{\sum_{k=1}^K \xi_{npk}^U} \quad (8.44)$$

$$\begin{cases} \mathbb{E}_p[l_{nk}] = \beta_{nk1} \beta_{nk2} \\ \mathbb{E}_p[\log l_{nk}] = \Psi(\beta_{nk1}) + \log \beta_{nk2} \end{cases} \quad (8.45)$$

However, the equations above involve three expectations w.r.t. the posterior distribution which cannot be solved analytically, and they require a moment matching algorithm. That is, the expectations are approximated by drawing samples from the posterior distribution through Gibbs sampling, which iteratively samples the complete conditionals in Eqs. (8.22)(8.23) until convergence. At that time, the samples from these conditionals corresponds to those of the true posterior  $x_{n::}^{(s)}, l_{n:}^{(s)} \sim p(x_{n::}, l_{n:} | y_{n:}, \Phi, p, r)$ . With these samples, one can approximate the expectations above as the following Monte Carlo estimates,

$$\mathbb{E}_p[x_{np:}] \approx \frac{1}{S} \sum_s x_{np:}^{(s)} \quad (8.46)$$

$$\mathbb{E}_p[l_{nk}] \approx \frac{1}{S} \sum_s l_{nk}^{(s)} \quad (8.47)$$

$$\mathbb{E}_p[\log l_{nk}] \approx \frac{1}{S} \sum_s \log l_{nk}^{(s)} \quad (8.48)$$

where  $S$  is the number of samples drawn with Gibbs sampling after an initial burn-in period of  $I$  iterations. Finally, these estimated expectations or moments can be used to find the variational parameters in Eqs. (8.43) (8.44) (8.45).

### 8.2.2.2 Approximating the Upper Bound

Given that Eq. (8.34) does not have an analytical closed-form solution, the upper-bound needs to be approximated. This can be done via a Monte Carlo sampler that reuses the same posterior samples used to compute the moments in Eqs. (8.46) (8.47) (8.48). That is,

$$\hat{U}_U \approx \frac{1}{S} \sum_{s=1}^S \log \frac{p(y_{n:}, x_{n:}^{(s)}, l_{n:}^{(s)} | \Phi, p, r)}{Q_{U_{y_{n:}}}(x_{n:}^{(s)}, l_{n:}^{(s)} | \gamma_{np:}^U, \beta_{nk:}^U)} \quad (8.49)$$

where  $x_{n:}^{(s)}, l_{n:}^{(s)} \sim p(x_{n:}, l_{n:} | y_{n:}, \Phi, p, r)$  and  $Q_{U_{y_{n:}}}(x_{n:}^{(s)}, l_{n:}^{(s)} | \gamma_{np:}^U, \beta_{nk:}^U)$  is given by Eq. (8.35).

### 8.2.2.3 Moment matching Algorithm

In Algorithm 8.2, we present the pseudocode to compute the parameters for the upper-bounded proposal  $Q_U(x_{n:}, l_{n:}; \gamma_n^U, \beta_n^U)$  as well as the approximated upper bound  $\hat{U}_U$ , based on the procedure above. The algorithm returns the variational parameters  $\gamma_{n:}^U$  which are used in the VIS estimator defined in Eq. (8.18), but the algorithm also computes  $\beta_n^U$ , which are used to approximate the EUBO. Note the Gibbs sampling algorithm is linear in the total number of Gibbs cycles  $I + S$  and in the number of non-zeros words  $V_{c_n}$  and topics  $K$ . Besides, the computation of the variational parameters is linear in the number of words and topics and for  $\beta_n^U$  also in the number of iterations of the Newton's method.

## 8.3 Mean-field VIS for BPFA

As we showed in Section 2.4.6, the augmented model for BPFA involves two discrete latent variables, the word counts  $y_{n:}$  and the topic counts  $x_{n:}$  and one continuous variable, the factors  $l_{n:}$ . If we collapse the continuous factors as in PFA, the two count variables, one of them with infinite support, have to be sampled with the IS. In the previous chapter, we showed that the sampling of these two variables leads to poor performance of the left-to-right sampler, specially for long-text. Besides we also experimented with this configuration for VIS and the convergence was slow too. Therefore, we present the VIS sampler for BPFA, which collapses both discrete variables and samples the continuous  $l_{n:}$  factors. The marginal document likelihood for the collapsed BPFA can be written as,

$$p(b_{n:}; \Phi, \mathbf{p}, \mathbf{r}) = \int p(b_{n:}, l_{n:}; \Phi, \mathbf{p}, \mathbf{r}) dl_{n:} \quad (8.50)$$

where the joint distribution  $p(b_{n:}, l_{n:}; \Phi, \mathbf{p}, \mathbf{r})$  is given by,

$$p(b_{n:}, l_{n:}; \Phi, \mathbf{p}, \mathbf{r}) = \prod_{p=1}^P \text{Ber}(b_{np}; 1 - e^{-l_{n:} \phi_{:p}}) \prod_{k=1}^K \text{Ga}(l_{nk}; r_k, \frac{p_k}{1 - p_k}) \quad (8.51)$$

according to the BerPo link property in Eq. (2.34).

**Algorithm 8.2:** PFA Upper-bounded Proposal.

---

```

input  :  $I, S, y_{n:}, \mathbf{r}, \mathbf{p}, \Phi$ 
output:  $\gamma_{n::}^U$ 

1  $V_{c_n} \leftarrow |y_{n:} > 0|$ 
2  $K \leftarrow \text{Length}(\mathbf{p})$ 
3 Function GibbsSampling( $I, S, y_{n:}, \mathbf{r}, \mathbf{p}, \Phi$ )
4   for  $k \leftarrow 1$  to  $K$  do
5      $l_{nk}^{(0)} \leftarrow \text{Init}(r_k, p_k)$ 
6   for  $i \leftarrow 1$  to  $I + S$  do
7     for  $p \leftarrow 1$  to  $V_{c_n}$  do
8        $x_{np}^{(i)} \leftarrow \text{MultinomialSampling}(y_{np}, \Phi, l_{n:}^{(i-1)})$  Eq. (8.22)
9       for  $k \leftarrow 1$  to  $K$  do
10         $l_{nk}^{(i)} \leftarrow \text{GammaSampling}(r_k, p_k, x_{n:k}^{(i)})$  Eq. (8.23)
11   return  $x_{n::}^{(I:(S+I))}, l_{n:}^{(I:(S+I))}$ 

12  $x_{n::}^{(\cdot)}, l_{n:}^{(\cdot)} \leftarrow \text{GibbsSampling}(I, S, y_{n:}, \mathbf{r}, \mathbf{p}, \Phi)$ 
13  $\mathbb{E}_p[x_{np:}] \leftarrow \text{ApproxExpX}(x_{n::}^{(\cdot)})$  Eq. (8.46)
14  $\mathbb{E}_p[l_{n:}] \leftarrow \text{ApproxExpL}(l_{n:}^{(\cdot)})$  Eq. (8.47)
15  $\mathbb{E}_p[\log l_{n:}] \leftarrow \text{ApproxExpLogL}(l_{n:}^{(\cdot)})$  Eq. (8.48)

16 for  $p \leftarrow 1$  to  $V_{c_n}$  do
17   for  $k \leftarrow 1$  to  $K$  do
18      $\xi_{npk}^U \leftarrow \text{ComputeXi}(\mathbb{E}_p[x_{npk}])$  Eq. (8.43)
19    $\gamma_{np:}^U \leftarrow \text{ComputeGamma}(\xi_{np:}^U)$  Eq. (8.44)

20 for  $k \leftarrow 1$  to  $K$  do
21    $\beta_{nk:}^U \leftarrow \text{ComputeBeta}(\mathbb{E}_p[l_{nk}], \mathbb{E}_p[\log l_{nk}])$  Eq. (8.45)

22  $\hat{U}_U \leftarrow \text{ApproximateEUBO}(y_{n:}, \mathbf{r}, \mathbf{p}, \Phi, x_{n::}^{(\cdot)}, l_{n:}^{(\cdot)}, \gamma_{n::}^U, \beta_{nk:}^U)$  Eq. (8.49)

```

---

As a result, we can build an IS estimator for Eq. (8.50) that samples  $l_{n:}$  from a proposal  $Q(l_{n:}; \beta_n)$  with support in  $\mathbb{R}_{>0}^K$  and computes a weighted average as follows,

$$p(b_{n:}; \Phi, \mathbf{p}, \mathbf{r}) \approx \frac{1}{S} \sum_{s=1}^S p(b_{n:} | l_{n:}^{(s)}; \Phi) w(l_{n:}^{(s)}) \quad \text{where } l_{n:}^{(s)} \sim Q(l_{n:}; \beta_n) \quad (8.52)$$

where the importance weights are given by  $w(l_{n:}^{(s)}) = \frac{p(l_{n:}^{(s)}; \mathbf{p}, \mathbf{r})}{Q(l_{n:}^{(s)}; \beta_n)}$ . The proposal  $Q(l_{n:}; \beta_n)$  can again be any variational distribution derived from the minimisation of KL divergences above. We next described how to derive mean-field distributions  $Q(l_{n:}; \beta_n)$  that minimises the KL divergences through the optimization of the upper and lower bounds of the marginal document likelihood, as in PFA.

However, the complete conditionals of the collapsed BPFA model are not in the exponential family and hence, analytical mean-field distributions cannot be derived from them.

Similar to what happened for PFA, we address this issue by learning the mean-field distributions for the augmented model in Section 2.4.6, and then, use the mean-field distributions associated to  $l_{n:}$  as proposal  $Q(l_{n:}; \beta_n)$  for the VIS estimator in Eq. (8.52).

### 8.3.1 Lower-bounded Mean-field Proposal

We can write the ELBO in Eq. (8.7) for the augmented BPFA model as follows,

$$\begin{aligned} U_L^B &= \mathbb{E}_{Q_L(y_{n:}, x_{n:}, l_{n:}; \lambda_n^L, \gamma_n^L, \beta_n^L)} \log \frac{p(b_{n:}, y_{n:}, x_{n:}, l_{n:}; \Phi, \mathbf{p}, \mathbf{r})}{Q_L(y_{n:}, x_{n:}, l_{n:}; \lambda_n^L, \gamma_n^L, \beta_n^L)} \\ &= \mathbb{E}_{Q_L(y_{n:}, x_{n:}, l_{n:}; \lambda_n^L, \gamma_n^L, \beta_n^L)} \log p(b_{n:}, y_{n:}, x_{n:}, l_{n:}; \Phi, \mathbf{p}, \mathbf{r}) + \mathbb{H}(y_{n:}, x_{n:}, l_{n:}; \lambda_n^L, \gamma_n^L, \beta_n^L) \end{aligned} \quad (8.53)$$

where the joint distribution for the observed indicator  $b_{n:}$ , the latent word counts  $y_{n:}$ , topic counts  $x_{n:}$  and factors  $l_{n:}$  are given by the same Eq. (8.20), except that the support for the word counts  $y_{n:}$  is also restricted in  $\mathbb{Y}_{b_{n:}} = \{y_{n:} \in \mathbb{N}_0^V \mid b_{n:} = \mathbf{1}(y_{n:})\}$ . We propose to use a structured mean-field proposal such as,

$$Q_L(y_{n:}, x_{n:}, l_{n:}; \lambda_n^L, \gamma_n^L, \beta_n^L) = Q_L(l_{n:}; \beta_n^L) \prod_{p=1}^V Q_{L_p}(y_{np}, x_{np}; \lambda_{np}^L, \gamma_{np}^L) \quad (8.54)$$

where  $Q_L(l_{n:}; \beta_n^L)$  is the same mean-field approximation than in PFA given by the product of the  $K$  distribution in Eq. (8.21), and the mean-field associated with the counts  $y_{n:}$  and  $x_{n:}$  is defined on  $\mathbb{Y}_{b_{n:}} \times \mathbb{X}_{y_{n:}}$ . Given that  $b_{np} = 0$  implies that both  $y_{np}$  and  $x_{np}$  are 0, one only needs to define the variational distribution for the non-zero cases  $p = 1 \dots V_{c_n}$  (we assume document are ordered such that all non-zeros precedes zeros).

The complete conditional for each factor  $l_{nk}$  is the same than in PFA, and it is given by Eq. (8.23). In contrast, the jointly conditional distribution for  $y_{np}, x_{np}$  in a non-zero word  $p$  is expressed as follows,

$$p(y_{np}, x_{np} | b_{np}, l_{n:}; \Phi, \mathbf{p}, \mathbf{r}) = \text{Pois}_+(y_{np}; \sum_{k=1}^K l_{nk} \phi_{kp}) \text{Mult}(x_{np}; y_{np}, \frac{l_{n:} \phi_{:p}}{\sum_{k=1}^K l_{nk} \phi_{kp}}) \quad (8.55)$$

which can be derived from the quotient of the joint distribution  $p(y_{np}, x_{np}, b_{np}, l_{n:}; \Phi, \mathbf{p}, \mathbf{r})$  given by Eq. (8.20) and its marginal  $p(b_{np}, l_{n:}; \Phi, \mathbf{p}, \mathbf{r})$  in Eq. (8.51). We note that the pmf for this distribution is given by,

$$\text{Pois}_+ \text{Mult}(x_{:}; \xi_{:}) = \frac{1}{e^{\sum_k \xi_k} - 1} \prod_k \frac{\xi_k^{x_k}}{x_k!} \quad (8.56)$$

where  $\xi_{:} = l_{n:} \phi_{:p}$  and  $x_{:} = x_{np}$  in Eq. (8.55). Note that this distribution is also in the exponential family with base measure  $h(x) = \frac{1}{\prod_k x_k!}$ , sufficient statistics  $t(x) = [x_1, \dots, x_K]$ , natural parameters  $\eta(\xi) = [\log \xi_1, \dots, \log \xi_K]$  and cumulant or log-partition function  $A(\eta(\xi)) = \log(e^{\sum_k \xi_k} - 1)$ . This is the distribution for  $K$  Poisson random variables, whose  $K$  realizations cannot be all 0 at the same time.

Therefore, the mean-field distribution for both  $y_{np}, x_{np}$  has the same analytical form than its conditional, and hence it is independent of  $y_{np}$ . Thus, we can write the mean-field distribution for  $y_{np}, x_{np}$  in Eq. (8.54) as,

$$Q_{L_p}(y_{np}, x_{np}; \lambda_{np}^L, \gamma_{np}^L) = Q_{L_p}(x_{np}; \xi_{np}^L) = \text{Pois}_+ \text{Mult}(x_{np}; \xi_{np}^L) \quad (8.57)$$

where  $\xi_{np}^L$  is the variational parameter for the  $p$ -th non-zero word. We can derive closed-form expressions for these parameters by taking the expectation of the natural parameters of the conditional in Eq. (8.55) as follows,

$$\eta(\xi_{np}^L) = \mathbb{E}_{Q_L(l_{n:}; \beta_n^L)}[\log l_{n:}] + \log \phi_{:p} \quad (8.58)$$

where the expectation of  $\log l_{n:}$  can be computed for each factor according to Eq. (8.27), and then express the variational parameters in their original parametrisation as,

$$\xi_{np}^L = \phi_{:w} \beta_{n:2}^L e^{\Psi(\beta_{n:1}^L)}. \quad (8.59)$$

We note that for the variational parameters of  $Q_L(l_{n:}; \beta_n^L)$ , we can follow the same derivation than in PFA,

$$\eta(\beta_{nk}^L) = \mathbb{E}_{Q_L(x_{n:}; \gamma_n^L)}[\eta(l_{nk})] = \left[ r_k + \sum_{p=1}^V \mathbb{E}_{Q_{L_p}(x_{np}; \xi_{np}^L)}[x_{npk}] - 1 \right] \quad (8.60)$$

except that the expectation is now w.r.t.  $Q_{L_p}(x_{np}; \xi_{np}^L)$ . This expectation can be calculated from the partial derivatives of the cumulant of the  $\text{Pois}_+ \text{Mult}(\cdot)$  and it is given in Eq. (E.22). Therefore, the variational parameters are as follows

$$\beta_{nk1}^L \leftarrow r_k + \sum_{p=1}^V \frac{\xi_{npk}}{1 - e^{-\sum_{k=1}^K \xi_{npk}}} \quad \beta_{nk2}^L \leftarrow p_k \quad (8.61)$$

The ELBO for BPFA in Eq. (8.53) can be derived from that of PFA by noting that only those terms that involve  $x_{np}$  must be rederived because their expectations are taken w.r.t. another distribution  $Q_{L_p}(x_{np}; \xi_{np}^L)$ . Therefore, the entropy term in Eq. (8.32) associated with  $x_{np}$  is derived in Appendix E.5; and similarly, the derivation of the expectation of  $\log p(x_{n:}, l_{n:}; \Phi, \mathbf{p}, \mathbf{r})$  in Eq. (8.33) can be found in Appendix E.6.

In Algorithm 8.3, we highlight in blue the main changes with respect to the same algorithm for PFA and line through an update that is no more required. Structurally, the algorithm is the same with only minor differences on the updates and expectations derived above. However, the lower-bounded mean-field proposal in Eq. (8.52) is constructed from the  $\beta_{n:}^L$  variational parameters, and hence the algorithm returns them instead. We also note that the observed data  $b_{n:}$  is only used to determine the last present word  $V_{c_n}$  in the ordered document.

### 8.3.2 Upper-bounded Mean-field Proposal

We can express the EUBO in Eq. (8.13) for the augmented BPFA as,

$$\begin{aligned} U_U^B &= \mathbb{E}_{p(y_{n:}, x_{n:}, l_{n:}; b_{n:}; \Phi, \mathbf{p}, \mathbf{r})} \log \frac{p(b_{n:}, y_{n:}, x_{n:}, l_{n:}; \Phi, \mathbf{p}, \mathbf{r})}{Q_U(y_{n:}, x_{n:}, l_{n:}; \lambda_n^U; \gamma_n^U, \beta_n^U)} \\ &= \mathbb{E}_{p(y_{n:}, x_{n:}, l_{n:}; b_{n:}; \Phi, \mathbf{p}, \mathbf{r})} \log p(b_{n:}, y_{n:}, x_{n:}, l_{n:}; \Phi, \mathbf{p}, \mathbf{r}) - \log Q_U(y_{n:}, x_{n:}, l_{n:}; \lambda_n^U; \gamma_n^U, \beta_n^U) \end{aligned} \quad (8.62)$$

**Algorithm 8.3:** BPFA Lower-bounded Proposal.

---

```

input  :  $b_{n:}, r, p, \Phi$ 
output:  $\beta_{n:}^L$ 

1  $V_{c_n} \leftarrow |b_{n:} > 0|$ 
2  $K \leftarrow \text{Length}(p)$ 
3 Function  $\text{ComputeELBO}(r, p, \Phi, \xi_{n:}^L, \beta_{n:}^L)$ 
4    $U^L \leftarrow 0$ 
5   for  $p \leftarrow 1$  to  $V_{c_n}$  do
6      $U^L \leftarrow U^L + \text{EntropyZTPs}(\xi_{np}^L)$  Eq. (E.24)
7   for  $k \leftarrow 1$  to  $K$  do
8      $U^L \leftarrow U^L + \text{EntropyGa}(\beta_{nk}^L)$  Eq. (E.12)
9      $U^L \leftarrow U^L + \text{ExpLogGa}(r_k, p_k, \beta_{nk}^L)$  Eq. (E.19)
10    for  $p \leftarrow 1$  to  $V_{c_n}$  do
11       $U^L \leftarrow U^L + \text{ExpLogPois}(\phi_{kp}, \xi_{np}^L, \beta_{nk}^L)$  Eq. (E.27)
12  return  $U^L$ 

/* Coordinate ascent algorithm */
13 for  $k \leftarrow 1$  to  $K$  do
14    $\beta_{nk}^L \leftarrow \text{Init}(r_k, p_k)$ 
15 while  $U_L \leftarrow \text{ComputeELBO}(r, p, \Phi, \xi_{n:}^L, \beta_{n:}^L)$  not converged do
16   for  $p \leftarrow 1$  to  $V_{c_n}$  do
17     for  $k \leftarrow 1$  to  $K$  do
18        $\xi_{npk}^L \leftarrow \text{UpdateXi}(\phi_{kp}, \beta_{nk}^L)$  Eq. (8.59)
19        $\gamma_{np}^L \leftarrow \text{UpdateGamma}()$ 
20     for  $k \leftarrow 1$  to  $K$  do
21        $\beta_{nk}^L \leftarrow \text{UpdateBeta}(\xi_{np}^L)$  Eq. (8.61)

```

---

where the joint distribution is again given by Eq. (8.20) with support for  $y_{n:} \in \mathbb{Y}_{b_{n:}}$ . Similar to PFA, we assume the same statistical forms for the mean-field distributions as the ones derived for the lower bound of BPFA. Thus, we reduce the variational distribution  $Q_U(y_{n:}, x_{n:}, l_{n:}; \lambda_{n:}^U, \gamma_{n:}^U, \beta_{n:}^U)$  to a distribution for the non-zero words that is function of  $x_{n:}$  and  $l_{n:}$ , as expressed by,

$$Q_U(x_{n:}, l_{n:}; \xi_{n:}^U, \beta_{n:}^U) = \prod_{k=1}^K \text{Ga}(l_{n:}; \beta_{n:}^U) \prod_{p=1}^{V_{c_n}} \text{Pois}_+ \text{Mult}(x_{np}; \xi_{np}^U) \quad (8.63)$$

where  $\text{Pois}_+ \text{Mult}(\cdot)$  is the pmf given in Eq. (8.56) and  $\text{Ga}(\cdot)$  is the pdf of a Gamma distribution given in Eq. (B.9)

Because of the factorisation across variables, parameters can be derived independently by applying Eq. (8.38) to the variational distribution above. Therefore, the variational parameters for  $\beta_{n:}^U$  derived in Eq. (8.45) are also valid for BPFA, but with expectations taken w.r.t.  $p(y_{n:}, x_{n:}, l_{n:}; |b_{n:}; \Phi, p, r)$ . In addition, because  $\text{Pois}_+ \text{Mult}(\cdot)$  is in the exponential

family, a global minimum for  $\xi_{np\cdot}$  also exists and can be found by solving the equation,

$$\mathbb{E}_{p(y_{n\cdot}, x_{n\cdot\cdot}, l_{n\cdot} | b_{n\cdot}; \Phi, \mathbf{p}, \mathbf{r})}[t(x_{np\cdot})] \frac{\delta \eta(\xi_{np\cdot})}{\delta \xi_{np\cdot}} - \frac{\delta A(\eta(\xi_{np\cdot}))}{\delta \xi_{np\cdot}} = 0 \quad p = 1 \dots V \quad (8.64)$$

which, by substituting  $\eta(\xi_{np\cdot}) = \log \xi_{np\cdot}$ ,  $A(\eta(\xi_{np\cdot})) = \log(e^{\xi_{np\cdot}} - 1)$  and deriving w.r.t.  $\xi_{np\cdot}$ , it can be expressed as,

$$\begin{bmatrix} \mathbb{E}_{p'}[x_{np1}] \\ \vdots \\ \mathbb{E}_{p'}[x_{npK}] \end{bmatrix} \begin{bmatrix} \frac{1}{\xi_{np1}} \\ \vdots \\ \frac{1}{\xi_{npK}} \end{bmatrix} - \begin{bmatrix} \frac{\xi_{np1}}{1 - e^{-\sum_{k=1}^K \xi_{npk}}} \\ \vdots \\ \frac{\xi_{npK}}{1 - e^{-\sum_{k=1}^K \xi_{npk}}} \end{bmatrix} = 0 \quad p = 1 \dots V_{c_n}. \quad (8.65)$$

where  $\mathbb{E}_{p'}$  refers to  $\mathbb{E}_{p(y_{n\cdot}, x_{n\cdot\cdot}, l_{n\cdot} | b_{n\cdot}; \Phi, \mathbf{p}, \mathbf{r})}$ . That is, we need to solve the equation for every factor  $k = 1 \dots K$  and non-zero word  $p = 1 \dots V_{c_n}$ .

$$\mathbb{E}_{p'}[x_{npk}] = \frac{\xi_{npk}^2}{1 - e^{-\sum_{k=1}^K \xi_{npk}}}. \quad (8.66)$$

Because this expression does not have an analytical solution, we use the Newton's method to iteratively solve it. As before, the expectations are approximated with Monte Carlo estimates, such as,

$$\mathbb{E}_{p'}[x_{npk}] = \frac{1}{S} \sum_{s=1}^S x_{npk}^{(s)}, \quad \text{where } x_{npk}^{(s)} \sim p(y_{n\cdot}, x_{n\cdot\cdot}, l_{n\cdot} | b_{n\cdot}; \Phi, \mathbf{p}, \mathbf{r}) \quad (8.67)$$

where the  $x_{npk}^{(s)}$  samples comes from the posterior of BPFA. Similarly, the expectations of  $l_{nk}$  and for  $\log l_{nk}$  in Eq. (8.45) must be computed w.r.t. this new posterior.

As a result, the algorithm to find the upper-bounded proposal for BPFA requires to sample the posterior distribution. We propose to use a Gibbs sampling algorithm that, apart from sampling the complete conditionals given by Eqs. (8.23) (8.22), it also samples the complete conditionals of  $y_{n\cdot}$  for the non-zero words  $b_{np} = 1$  as follows,

$$p(y_{np} | l_{n\cdot}; \Phi, \mathbf{p}, \mathbf{r}) = \text{Pois}_+(y_{np}; \sum_{k=1}^K l_{nk} \phi_{kp}) \quad (8.68)$$

With these posterior samples, we can compute a Monte Carlo estimate of the upper bound for BPFA as,

$$\hat{U}_U^B \approx \frac{1}{S} \sum_{s=1}^S \log \frac{p(b_{n\cdot}, y_{n\cdot}^{(s)}, x_{n\cdot\cdot}^{(s)}, l_{n\cdot}^{(s)}; \Phi, \mathbf{p}, \mathbf{r})}{Q_U(x_{n\cdot\cdot}^{(s)}, l_{n\cdot}^{(s)}; \xi_{n\cdot\cdot}^U, \beta_{n\cdot\cdot}^U)}, \quad \text{where } y_{n\cdot}^{(s)}, x_{n\cdot\cdot}^{(s)}, l_{n\cdot}^{(s)} \sim p(y_{n\cdot}, x_{n\cdot\cdot}, l_{n\cdot} | b_{n\cdot}; \Phi, \mathbf{p}, \mathbf{r}). \quad (8.69)$$

In Algorithm 8.4, we present the pseudocode for deriving the upper-bounded mean-field proposal for BPFA. In blue, we also highlight the main differences w.r.t. the algorithm developed for PFA. We observe that these are mainly in the Gibbs Sampling scheme, where the word counts need to be sampled, and in the computation of the variational parameters  $\xi_{npk}^U$ , where there is different update formula. Moreover, we also note that the expectation

are computed w.r.t. another posterior distribution. Finally, as in the lower-bound proposal for BPFA, we note that the observed variables  $b_{n:}$  are only involved in determining the number of non-zero words, as long as the document is sorted such that all non-zeros precedes zero words.

---

**Algorithm 8.4:** BPFA Upper-bounded Proposal.

---

```

input :  $I, S, \mathbf{r}, \mathbf{p}, \Phi$ 
output:  $\beta_{n::}^U$ 

1  $V_{c_n} \leftarrow |b_{n:} > 0|$ 
2  $K \leftarrow \text{Length}(\mathbf{p})$ 
3 Function GibbsSampling( $I, S, \mathbf{r}, \mathbf{p}, \Phi$ )
4   for  $k \leftarrow 1$  to  $K$  do
5      $l_{nk}^{(0)} \leftarrow \text{Init}(r_k, p_k)$ 
6   for  $i \leftarrow 1$  to  $I + S$  do
7     for  $p \leftarrow 1$  to  $V_{c_n}$  do
8        $y_{np}^{(i)} \leftarrow \text{Pois}_+\text{Sampling}(\Phi, l_{n:}^{(i-1)})$ 
9        $x_{np:}^{(i)} \leftarrow \text{MultinomialSampling}(y_{n:}^{(I:(S+I))}, \Phi, l_{n:}^{(i-1)})$       Eq. (8.22)
10      for  $k \leftarrow 1$  to  $K$  do
11         $l_{n:k}^{(i)} \leftarrow \text{GammaSampling}(r_k, p_k, x_{n:k}^{(i)})$       Eq. (8.23)
12    return  $x_{n::}^{(I:(S+I))}, l_{n:}^{(I:(S+I))}$ 

13  $x_{n::}^{(\cdot)}, l_{n:}^{(\cdot)} \leftarrow \text{GibbsSampling}(I, S, \mathbf{r}, \mathbf{p}, \Phi)$ 
14  $\mathbb{E}_{p'}[x_{np:}] \leftarrow \text{ApproxExpX}(x_{n::}^{(\cdot)})$       Eq. (8.46)
15  $\mathbb{E}_{p'}[l_{nk}] \leftarrow \text{ApproxExpL}(l_{n:}^{(\cdot)})$       Eq. (8.47)
16  $\mathbb{E}_{p'}[\log l_{nk}] \leftarrow \text{ApproxExpLogL}(l_{n:}^{(\cdot)})$       Eq. (8.48)
17 for  $p \leftarrow 1$  to  $V_{c_n}$  do
18   for  $k \leftarrow 1$  to  $K$  do
19      $\xi_{npk}^U \leftarrow \text{ComputeXi}(\mathbb{E}_{p'}[x_{npk}])$       Eq. (8.66)
20      $\gamma_{np:}^U \leftarrow \text{ComputeGamma}()$ 
21 for  $k \leftarrow 1$  to  $K$  do
22    $\beta_{nk:}^U \leftarrow \text{ComputeBeta}(\mathbb{E}_{p'}[l_{nk}], \mathbb{E}_{p'}[\log l_{nk}])$       Eq. (8.45)
23  $\hat{U}_U \leftarrow \text{ApproximateEUBO}(\mathbf{r}, \mathbf{p}, \Phi, x_{n::}^{(\cdot)}, l_{n:}^{(\cdot)}, \xi_{n::}^U, \beta_{n::}^U)$       Eq. (8.69)

```

---

## 8.4 Experimentation

In this section, we evaluate the accuracy and convergence properties of the VIS estimators presented earlier. We follow the experimental setup described in Section 6.3: we evaluate the accuracy of PFA estimators against the exact marginal likelihood in tractable scenarios, and we assess the convergence of PFA and BPFA estimators in realistic document collections. For both PFA and BPFA, we train the same non-parametric models described in Section 7.3.1 and compare the VIS estimators with the L2R, DS (Direct Sampling) and HM (Harmonic Mean) methods studied in the previous chapter. Furthermore, we use the lower



$(U_L, U_L^B)$  and upper  $(\hat{U}_U, \hat{U}_U^B)$  bounds derived earlier to sandwich the marginal document likelihood and discuss the accuracy of the candidate methods in realistic conditions.

### 8.4.1 Experiments with PFA in Downsized Collections

We first evaluate the VIS estimators developed for PFA in tractable scenarios. We use the downsized collections presented in Section 6.3.3 and train a  $\beta\Gamma$ -PFA model (Zhou et al., 2012) with hyperparameters set according to Table 7.1 and  $K_{max} = 5$ . Under these conditions, we compute the exact marginal document likelihood given by Eq. (6.7) for those documents that cause less than  $10^9$  partitions in the exact formula. With this quantity as a ground truth, we compare the estimates of the methods in the previous chapter (DS, HM, L2R and L2R with exact conditionals) against the lower-bounded (LB-VIS) and upper-bounded (UB-VIS) methods proposed here. More specifically, we compare the likelihood estimates of  $N$  documents against their exact likelihood in terms of KL divergence as described in Section 6.3.1.

Fig. 8.2 plots the KL divergences as a function of the number of samples used for each estimation method in the 6 downsized collections. As depicted in the plots, the UB-VIS (Upper-Bounded Variational Importance Sampling) method achieves the lowest KL divergence across all data sets. Furthermore, the LB-VIS (Lower-Bounded Variational Importance Sampling) method obtains the 2nd best position in 5 out of 6 corpora and its accuracy is comparable to that of the L2R methods in the NIPS collection. However, the comparison in the downsized NIPS collection might not be significant because it only contains one document with less than  $10^9$  partitions.

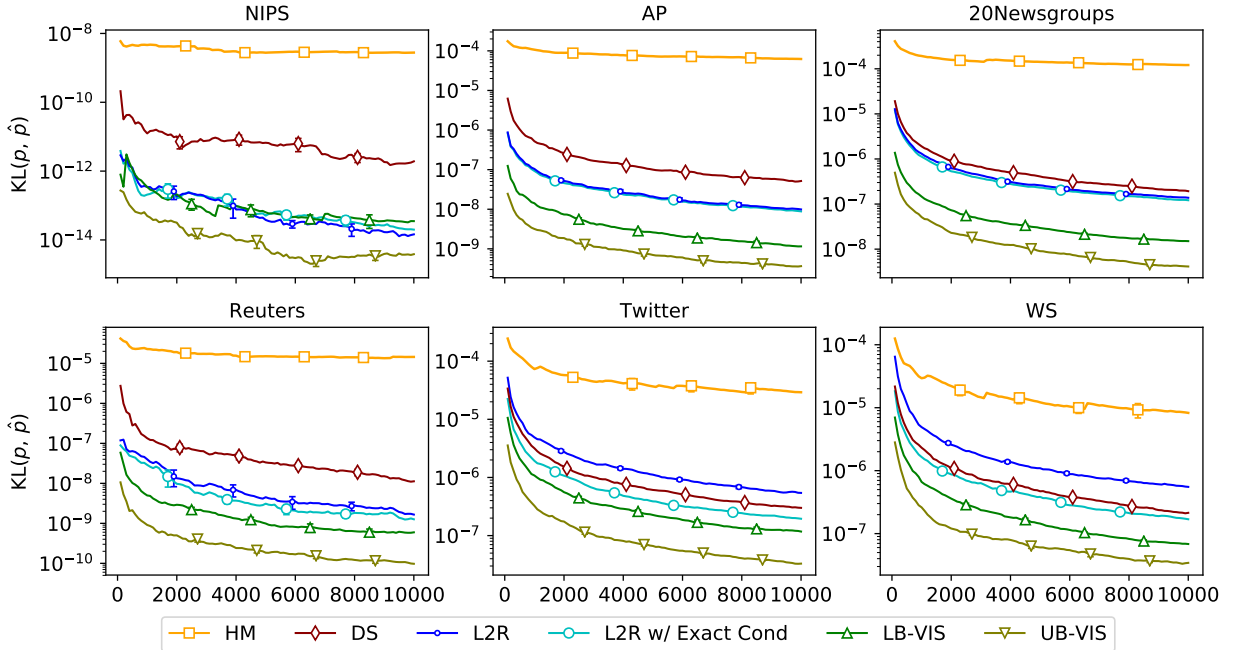


Figure 8.2: Relative Entropy or KL between the estimated document probabilities and the exacts as a function of samples used (Lower KL is better).

### 8.4.2 Experiments with PFA in Realistic Collections

We then evaluate the convergence properties of these estimators in the realistic collections reported in Table 6.2, by analysing the convergence in terms of the number of samples required by the log-likelihood to plateau. That is, we plot the marginal document log-likelihood of  $N = 1000$  documents as a function of the number of samples and we visually identify which methods stabilize faster than others.

In Fig. 8.3, we observe that the upper-bounded and lower-bounded VIS estimators reach convergences faster than the rest of candidates in 5 collections. Moreover, not all methods converge to the same values because of the different type of approximation considered by each method. In the WS data set, which is composed of the shortest documents, the convergence of the L2R is better than its competitors, followed closely by the UB-VIS and DS methods. We speculate that the slow convergence of the LB-VIS estimator could be related with the fact that the lower-bounded proposal is focused into one of the multiple modes of the posterior, that would explain why even the prior proposal used for the DS method can converge faster and to a higher log-likelihood than the LB-VIS. Furthermore, the fact that the convergence of the UB-VIS method is similar to that of DS suggests that the posterior distribution in this data set is spread across a wide space which cannot be precisely captured by the upper-bounded approximation. The spread of probability in short text data sets like WS or Twitter can be attributed to the troubles that topic models have in learning meaningful thematic structure in short text. More importantly, we note that the L2R algorithm from the previous chapter shows good performance in these data sets.

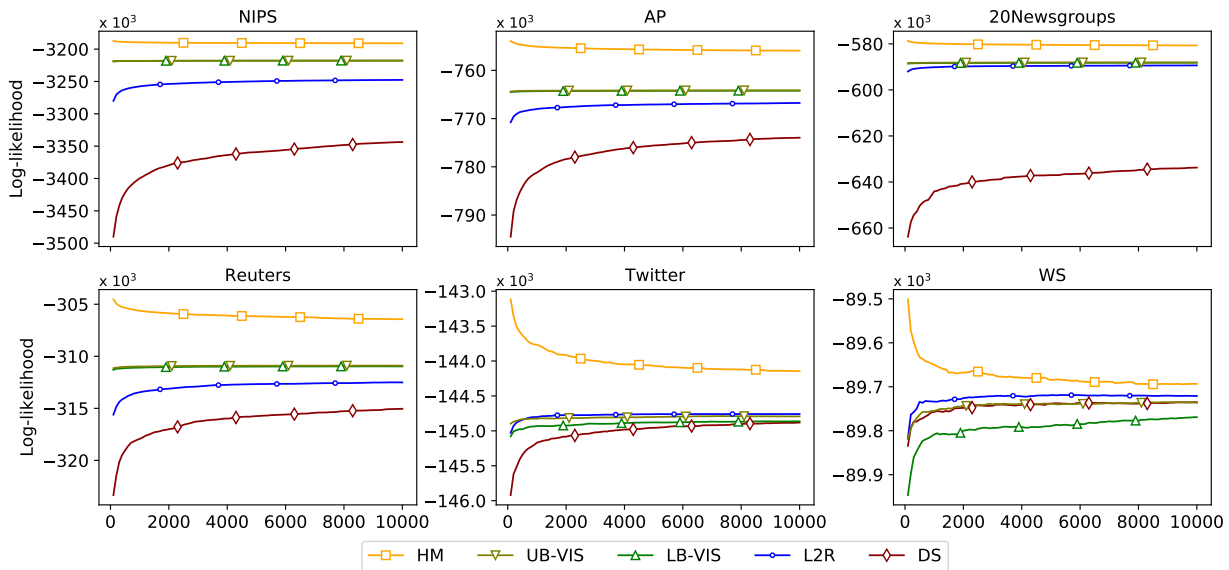


Figure 8.3: Document log-likelihood as a function of the number of samples in PFA.

In Fig. 8.4, we add the upper and lower bounds to the marginal document log-likelihoods from the previous figure. The lower bound or ELBO given by Eq. (8.19) can be assessed analytically in linear time in the number of topics and non-zero words, and hence it is a strong reference to determine which methods underestimate the log-likelihood. Whereas the upper bound or EUBO given by Eq. (8.49) is approximated via sampling, but we have observed a fast convergence in all setups and hence, it can be used to determine which meth-

ods might overestimate the log-likelihood. As it is shown in the figure, the VIS estimators developed in this chapter are the only methods sandwiched by both bounds. We note that the DS method underestimates the log-likelihood in the 4 long text collections (NIPS, AP, 20Newsgroups and Reuters) and the HM methods overestimates the log-likelihood in the AP corpus. We note, however, that both bounds are too loose in short text data sets (Twitter and WS) to be useful for determining the accuracy of these methods. The loose bounds also indicate us that the mean-field approximations might not be expressive enough to capture the complex posterior distribution in short text, and hence, why these mean-field proposals do not work in short text as well as in long text.

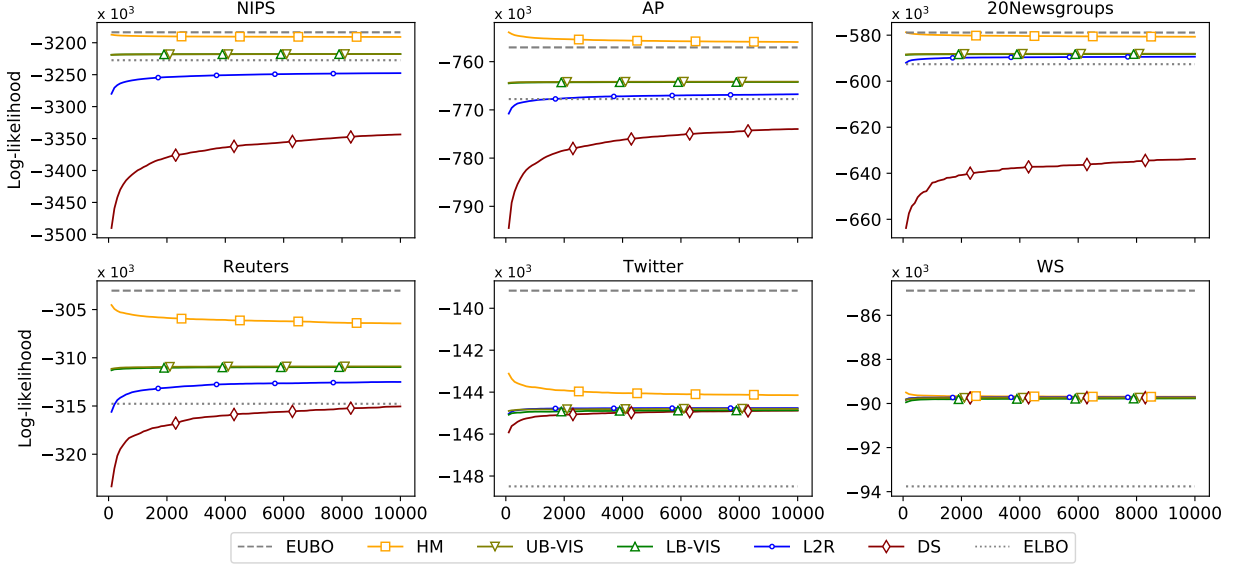


Figure 8.4: Sandwiched estimates as a function of the number of samples in PFA.

### 8.4.3 Experiments with BPFA in Realistic Collections

In this section, we evaluate the VIS estimators for BPFA developed earlier. For that, we train a  $\beta\gamma\Gamma$ -BPFA model (Zhou et al., 2012; Hu et al., 2016) with hyperparameters set according to Table 7.1. We evaluate them directly in realistic collections, because there is not a closed-form expression for computing the exact marginal document likelihood in finite time. In particular, we consider four binarised collections: two long text (NIPS and 20Newsgroups) and two short text (Twitter and WS).

In Fig. 8.5, we plot the marginal document log-likelihood as a function of the number of samples and observe the speed of convergence. For the long text data sets NIPS and 20Newsgroups, the L2R method was not plotted because the estimates were far below the rest as seen in the experimental results of the previous chapter. We notice that the UB-VIS's convergence is extremely fast in the 4 data sets. However, the convergence of LB-VIS was the worst across all corpora and comparatively worse than in PFA. We think that the posterior distribution is far more complex in BPFA than PFA, because of the latent word counts  $y_n$ . This would explain why the lower-bounded approximation, which is focused on a single mode, is even a worse proposal in the binarised scenarios.

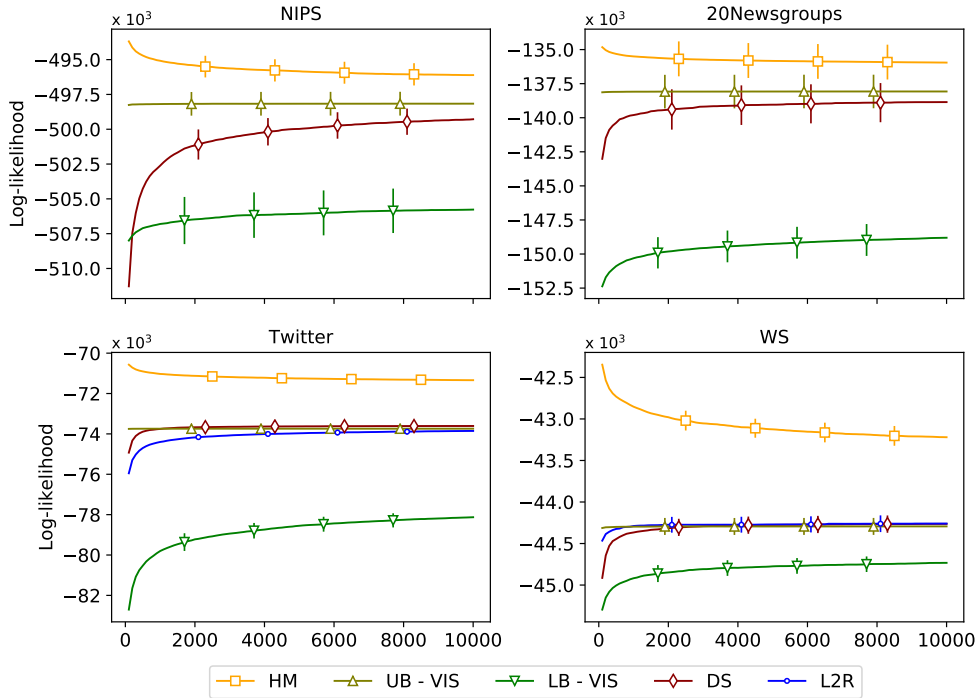


Figure 8.5: Document log-likelihood as a function of the number of samples in BPFA.

Finally, in Fig. 8.6 we add the upper and lower bounds derived in Eq. (8.53) and Eq. (8.62). Through these bounds, we can now confirm the inaccurate estimates obtained by the L2R algorithm in long text. Unfortunately, the bounds are too loose to determine the accuracy of the other methods.

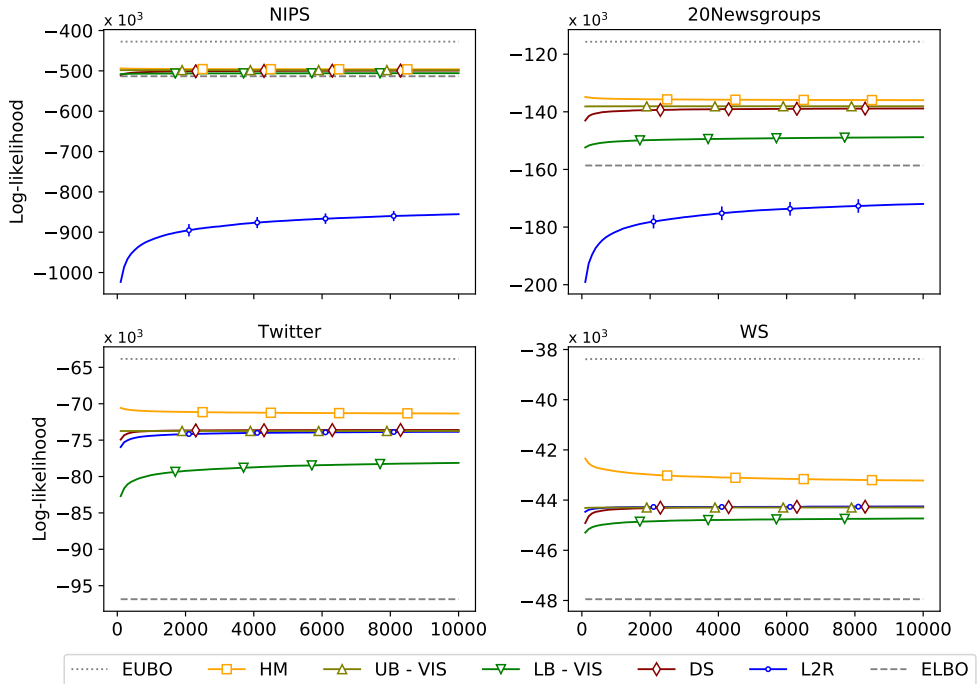


Figure 8.6: Sandwiched estimates as a function of the number of samples in BPFA.

## 8.5 Summary and Conclusion

In this chapter, we introduced a new class of estimators based on IS. In particular, we addressed the problem of finding proposal distributions which are close to the optimal in terms of KL divergence. We proposed a class of factorised distributions, known as the mean-field family, to construct proposal distributions for IS estimators. Furthermore, we showed that the minimisation of the KL divergences can result in two different types of solutions due to its asymmetry. Specially, we hypothesise that the minimisation of the forward KL divergence can lead to better proposals due to the fact that they are not centred at a specific posterior mode. Besides we showed that the minimization of the reverse KL and the forward KL lead respectively to an upper bound and a lower bound of the marginal log-likelihood, which can be useful to determine the accuracy of the estimation methods in realistic conditions.

We then derived the lower-bounded and upper-bounded mean-field proposals for the PFA and BPFA models. On the one hand, we showed that the lower-bounded proposals for PFA and BPFA can be found via coordinate ascent algorithms that iteratively update the variational parameters in a deterministic manner until convergence. The final values for the variational parameters can be also used to compute the lower bounds. On the other hand, we showed that the upper-bounded proposals for PFA and BPFA can be found by approximating several expectations with Monte Carlo sampling. With these expectations, the algorithms solve a system of equations to find the variational parameters in analytical manner (for  $\gamma_{np}^U$ ) and via the Newton's method (for  $\beta_{nk}^U$ ). Moreover, the posterior samples are used to approximate the upper bounds. It is important to note that these algorithms are linear in the number of topics and the number of non-zero words.

With these proposals, we then evaluated the VIS methods for both PFA and BPFA and compared their estimates to those of DS, HM and L2R in six document collections. For PFA, we showed that the proposed VIS methods are the most accurate in approximating the exact marginal in downsized collections. In more realistic setups, UB-VIS is always the best whilst LB-VIS has slower convergence in short text. Furthermore, we could determine the inaccuracy of some methods thanks to the bounds derived in this chapter. Although it is unsurprising that VIS methods are always sandwiched by the upper and lower variational bounds, we observed that the other methods were out of these bounds in some of the data sets. Besides, we could confirm that the DS and HM methods tend to under- and over- estimate the marginal log-likelihood, respectively. For BPFA, we studied the convergence and accuracy properties directly in realistic scenarios, because of the lack of a tractable closed-form expression to compare with the exact marginal. We showed that the immediate convergence of the UB-VIS method in long and short text and confirmed the underestimation of the L2R method in long text.

In conclusion, we proposed an upper-bounded mean-field variational importance sampling method, referred to as UB-VIS, that achieves the best accuracy in both short and long text for PFA and BPFA. Moreover, due to its fast convergence, this method can also produce highly accurate estimates with less samples than its competitors. Despite its higher computational cost, we also note that the L2R sequential sampler performed particularly well in short text, achieving faster convergence than its competitors in the WS collection.



# Part III

## Chordal Models





# Learning Chordal Models on Binarised Text

*“New ideas must use old buildings”*

JANE Jacobs

Capdevila, J., Zhao, H., Petitjean, F., and Buntine, W. (2018d). Experiments with learning graphical models on text. *Behaviormetrika*

In Part I of this dissertation, we used latent variable graphical models for uncovering events in short text. In particular, we showed that topics learned from pooled tweets helped the detection of events, and vice versa, pooling helped the semantic coherence of topics. Unfortunately, the prediction performance of LVM-based topic models in short text is known to deteriorate when contextual information is not available. This poor performance is attributed at the insufficient word co-occurrence to accurately estimate the local latent variables. In what follows, we consider a different class of graphical models which does not use local latent variables, but is capable of learning complex probability models for text through the statistical relationships in the global variables, i.e. words in the vocabulary.

The earliest forms of PGMs (Probabilistic Graphical Models) were BNs (Bayesian Networks) over discrete fully-observed variables, for which a huge variety of algorithms to learn their structure and parameters has already been developed (Heckerman and Chickering, 1995). Standard implementations, however, are usually restricted to less than 100 variables. More recently, improved data structures and algorithms have allowed models to be built with a larger number of variables. Branch and bound techniques allow best model search (Suzuki and Kawahara, 2017), but the use of memoization and restriction to chordal graphs (Petitjean and Webb, 2015b) have enabled the scaling to thousands of discrete variables. The algorithm for doing this is known as *Chordalysis* and it allows learning graphical

models on text corpora with vocabulary sizes in the order of thousands. However, *Chordalysis* was initially proposed for finding statistically significant associations and the models it learns are usually too simple to fit and generalise well on unseen data (i.e. prediction). Therefore, the Achilles’ heel of *Chordalysis* in prediction tasks lies in:

- Scoring function to learn the structure of the graphical model: [Webb and Petitjean \(2016\)](#) proposed the SMT (Subfamilywise Multiple Testing) metric which is sought to minimise the probability of false discoveries.
- Parameter estimation: *Chordalysis* uses maximum likelihood estimates to learn the model parameters, which might not be appropriate for more complex models with bigger cliques.

In this chapter, we propose more optimistic metrics (i.e. more prone to identify associations among variables), to learn the chordal structure of these models as well as suitable parameter estimation techniques for models with higher complexity. In particular, we show how to integrate the BIC (Bayesian Information Criterion) ([Schwarz et al., 1978](#)) and the qNML (quotient Normalized Maximum Likelihood) ([Silander et al., 2018](#)) with *Chordalysis* and we introduce three parameter estimation methods (two based on m-estimates and one, on Back-off estimates ([Friedman et al., 1997](#))) which mitigate the scarcity of data in big cliques. We show that these scoring functions and parameter estimates all achieve better prediction performance than the original methods in *Chordalysis* ([Petitjean and Webb, 2015b](#)).

To compare the new *Chordalysis* models with existing LVMs (Latent Variable Models) on text, we restrict the experimental study to models that use the binarised and bagged representation of text introduced in Chapter 2. In particular, we use the non-parametric Poisson factorisation presented in Section 2.4.6 as a representative for topic modelling and we modify it according to [Zhou \(2015\)](#) for binarised data. Another type of graphical model built on binarised data is the HLTA (Hierarchical latent tree analysis) ([Liu et al., 2014](#); [Chen et al., 2017](#)), which yields intriguing “local” topics, that only interact with a limited set of variables. The experiments are performed across several document collections with good representatives of short and long text, as well as big and small data. The results indicate that the amended *Chordalysis* models perform better in short text than existing topic models and they also scale to bigger datasets.

In Section 9.1, we review the original *Chordalysis* algorithm proposed in [Petitjean and Webb \(2015b\)](#); [Petitjean et al. \(2013\)](#) and introduce metrics and parameter estimation methods more suitable for prediction tasks. Then, we present the related work of graphical models for binarised text in Section 9.2. In Section 9.3, we discuss the experimental methodology. Finally, the results of experiments on binarised text collections are reported in Section 9.4.

## 9.1 The *Chordalysis* algorithm

The *Chordalysis* algorithm was initially proposed in [Petitjean and Webb \(2015b\)](#); [Petitjean et al. \(2013\)](#) to scale LLA (Log-Linear Analysis) to high-dimensional data. LLA is a well-established statistical technique for finding associations between several categorical variables and classical approaches consist in searching for a probability model  $\mathcal{M}^*$  that explains the

observed associations. The probability model can be expressed through a graph  $\mathcal{G} = (\mathcal{V}, \mathcal{E})$  and naively discovered through a forward selection algorithm in which a model is refined by adding edges one at a time until a specific metric (e.g.  $\chi^2$  goodness-of-fit) stops improving. However, the evaluation of one model over another in these approaches is exponential with respect to the number of variables and hence, they do not scale to more than a few variables.

*Chordalysis* addresses this issue by restricting to the sub-class of decomposable models, which have interesting computational properties. These are probabilistic graphical models whose supporting graph is chordal<sup>1</sup>. This class of models is not only practical for searching purposes, but it is also an expressive class of models, because a model in this class can subsume any non-decomposable model, and hence it can represent its statistical distribution. For instance, the saturated model, which is decomposable, subsumes any other graphical model.

As shown in Section 2.2.3, the probability mass function of a chordal model with  $V$  random variables  $X = \{X_1, \dots, X_V\}$  can be expressed in terms of its graphical structure as,

$$p(X; \theta, \mathcal{M}) = \frac{\prod_{c \in \mathcal{C}} p(X_c; \theta_c)}{\prod_{s \in \mathcal{S}} p(X_s; \theta_s)} \quad (9.1)$$

where  $\mathcal{C}$  represents the set of maximal cliques in the graph  $\mathcal{G}$  and  $\mathcal{S}$ , the set of minimal separators.  $\theta_c$  and  $\theta_s$  represent the sets of parameters corresponding to the distributions  $p(\cdot)$  of the cliques and separators, respectively. From this expression, [Petitjean et al. \(2013\)](#) re-wrote the likelihood ratio test statistic  $G^2$  for a chordal model as follows,

$$G^2(\mathcal{M}) = 2N \left( \sum_{c \in \mathcal{C}} \mathbb{H}(X_c) - \sum_{s \in \mathcal{S}} \mathbb{H}(X_s) - \mathbb{H}(X) \right) \quad (9.2)$$

where  $N$  is the total number of data points,  $\mathbb{H}(X_c)$  denotes the entropy of the variables in the maximal cliques,  $\mathbb{H}(X_s)$ , the entropy of the variables in the minimal separators and  $\mathbb{H}(X)$  is the empirical entropy, independent of  $\mathcal{M}$ .

The factorisation in Eq. (9.1) also enabled [Petitjean et al. \(2013\)](#) to rewrite the  $G^2$  statistic of a reference model  $\mathcal{M}^*$  versus a candidate model  $\mathcal{M}^c$ , which both are decomposable and only differ on the edge  $\{a, b\}$ , as,

$$G^2(\mathcal{M}^* \text{ v.s. } \mathcal{M}^c) = 2N \left( \mathbb{H}(X_{S_{ab} \cup \{a\}}) + \mathbb{H}(X_{S_{ab} \cup \{b\}}) - \mathbb{H}(X_{S_{ab} \cup \{a, b\}}) - \mathbb{H}(X_{S_{ab}}) \right) \quad (9.3)$$

where  $S_{ab}$  was the minimal separator between  $a$  and  $b$ ,  $\mathbb{H}(\cdot)$  denoted the entropy of the variables in the clique and  $N$  is the number of data points. As a result, the  $G^2$  statistic calculation to compare the current model with a model that incorporated the edge  $\{a, b\}$  could be reduced to the evaluation of four marginal entropies. Furthermore, they showed that the forward selection process entails many overlapping sub-problems which involve the evaluation of these entropies. Thus, they proposed the memoization of these partial solutions which were pre-computed through efficient counting on Tidsets, i.e. data structures for itemset counting ([Ogihara et al., 1997](#)). In [Petitjean and Webb \(2015b\)](#), authors identified that at every step of the forward process, only a subset of edges needed to be re-evaluated and proposed the *Prioritized Chordalysis* algorithm that identifies this subset efficiently.

---

<sup>1</sup>See definition in Section 2.2.3

Therefore, any metric that decomposes the scoring of two chordal models into the scoring of these 4 cliques ( $S_{ab} \cup \{a\}$ ,  $S_{ab} \cup \{b\}$ ,  $S_{ab} \cup \{a, b\}$ ,  $S_{ab}$ ) can be used in *Chordalysis*. Next, we will present new decomposable scoring functions suitable for prediction, which can be integrated with *Chordalysis* to learn more complex chordal models. These new models will lead to graphs with bigger cliques and hence, they will require more sophisticated estimation techniques to learn their parameters. Next, we will also introduce parameter estimation methods for smoothing the estimates in such situations.

### 9.1.1 Scoring Functions

The *Chordalysis* algorithm was initially proposed for uncovering statistical significant associations and hence, it makes extensive use of statistical testing through the  $G^2$  statistic. To control for the multiple hypothesis testing, which might lead to false discoveries, [Webb and Petitjean \(2016\)](#) introduced the SMT which was shown to be superior to the Bonferroni correction method and the layered critical values ([Webb, 2008](#)). However, the SMT score is too conservative for prediction purposes and other existing metrics such as the BIC ([Schwarz et al., 1978](#)), BDeu (Bayesian Dirichlet equivalent uniform) ([Buntine, 1991](#)) or qNML ([Silander et al., 2018](#)), seem more appropriate for this task.

#### 9.1.1.1 Bayesian Information Criterion (BIC)

For a model  $\mathcal{M}$  with  $V$  variables and a data collection  $\mathcal{D} = \{x_1, \dots, x_N\}$  with  $N$  observations, the BIC ([Schwarz et al., 1978](#)) is defined as,

$$\text{BIC}(\mathcal{M}) = -2 \log p(\mathcal{D}; \theta^{ML}, \mathcal{M}) + k \log N \quad (9.4)$$

where  $p(\mathcal{D}|\theta^{ML}; \mathcal{M})$  is the likelihood at its maximum, which is given by the maximum likelihood parameters  $\theta^{ML}$ , and  $k$  is the number of parameters used by the model. A model with lower BIC is preferred because it leads to higher likelihoods while penalizing the model complexity through the number of parameters.

Then, the BIC score between a reference model  $\mathcal{M}^*$  and a candidate model  $\mathcal{M}^c$  with an extra edge  $\{a, b\}$  can be expressed as,

$$\text{BIC}(\mathcal{M}^* \text{ v.s. } \mathcal{M}^c) = -2 \log \frac{p(\mathcal{D}; \theta^{ML}, \mathcal{M}^*)}{p(\mathcal{D}; \theta^{ML}, \mathcal{M}^c)} + (k^* - k^c) \log(N) \quad (9.5)$$

where  $p(\cdot)$  refers to the likelihood at its maximum for each model, and  $k^* - k^c$  is the difference on the number of parameters between models. The log ratio between the two likelihoods can be simplified as before by considering the corresponding cliques,

$$\begin{aligned} \log \frac{p(\mathcal{D}; \theta^{ML}, \mathcal{M}^*)}{p(\mathcal{D}; \theta^{ML}, \mathcal{M}^c)} &= \sum_{n=1}^N \log p(X_{S_{ab} \cup \{a\}} = x_{n, S_{ab} \cup \{a\}}) + \log p(X_{S_{ab} \cup \{b\}} = x_{n, S_{ab} \cup \{b\}}) \\ &\quad - \log p(X_{S_{ab} \cup \{a, b\}} = x_{n, S_{ab} \cup \{a, b\}}) - \log p(X_{S_{ab}} = x_{n, S_{ab}}) \end{aligned} \quad (9.6)$$

where  $\log p(\cdot)$  denotes the log probability of the variables in each clique. As shown in [Petitjean et al. \(2014\)](#), the difference in the parameters can also be simplified to the number of parameters of the 4 cliques above,

$$k^* - k^c = \text{param}(X_{S_{ab} \cup \{a\}}) + \text{param}(X_{S_{ab} \cup \{b\}}) - \text{param}(X_{S_{ab} \cup \{a, b\}}) - \text{param}(X_{S_{ab}}) \quad (9.7)$$

where  $\text{param}(\mathcal{A}) = -1 + \prod_{v \in \mathcal{A}} |\text{dom}(v)|$  and  $|\text{dom}(v)|$  is the number of outcomes of variable  $v$ .

However, BIC tends to require large sample sizes to recover the appropriate structure (Liu et al., 2012), as well as it has stronger bias for simpler models in both small and large sample sizes due to its pessimistic penalty term.

### 9.1.1.2 Quotient Normalised Maximum Likelihood (qNML)

Recently, the qNML (Silander et al., 2018) was proposed as an alternative for learning BN structures and shown to be more optimistic than BIC, but less than the BDeu. Moreover, qNML satisfies the score equivalence property (i.e. it produces equal scores to graphs that encode the same independences), is decomposable and is not sensitive to hyperparameters. Because of all this, we next detail how qNML is integrated with the Chordalysis algorithm.

The qNML was defined in (Silander et al., 2018) for a directed graphical model  $\mathcal{M}$  and the data collection  $\mathcal{D} = \{x_1, \dots, x_N\}$  in which each datum  $x_{n\cdot}$  is a  $V$ -dimensional vector and  $X_i$  refers to the  $i$ -th random variable. Then, the qNML is given by the following expression,

$$s^{qNML}(\mathcal{D}; \mathcal{M}) = \sum_{i=1}^V s_i^{qNML}(\mathcal{D}; \mathcal{M}) \quad (9.8)$$

$$= \sum_{i=1}^V \log \frac{p_{NML}^1(X_{i \cup \text{par}(i)})}{p_{NML}^1(X_{\text{par}(i)})} \quad (9.9)$$

where the score factorises across variables with each factor being the logarithm of the quotient of the one-dimensional NML (Normalized Maximum Likelihood), i.e.  $p_{NML}^1(\cdot)$ . The numerator of this ratio corresponds to the normalised likelihood for a single Multinomial variable with  $r = \prod_{t \in \{i \cup \text{par}(i)\}} r_t$  outcomes and the denominator is the normalised likelihood for a Multinomial variable with  $r = \prod_{t \in \{\text{par}(i)\}} r_t$  outcomes. Thus, the number of outcomes is calculated from the product of outcomes for each variable  $r_t$  in the corresponding set. The set in the numerator  $\{i, \text{par}(i)\}$  is composed of the  $i$ -th variable and its parents  $\text{par}(i)$ , whereas the denominator only contains the parents of variable  $i$ . Furthermore, each one-dimensional normalised likelihood for a set of variables  $S$  can be computed as,

$$p_{NML}^1(X_S) = \frac{p(X_S | \theta^{ML})}{\sum_{X'_S} p(X'_S | \theta^{ML})} \quad (9.10)$$

where  $\theta^{ML}$  are the maximum likelihood estimates for the parameters in the set  $S$  and the sum in the denominator goes over all possible data matrices  $X'_S \in \{1, \dots, r\}^N$ . Therefore, the numerator is simply the likelihood of the Multinomial distribution with  $r$  outcomes,  $N$  trials and parameters  $\theta^{ML} = \{\frac{n_1}{N}, \dots, \frac{n_r}{N}\}$ . The logarithm of the denominator, also called regret, can be either approximated with the formulas given in (Silander et al., 2018) or computed exactly for fixed  $N$  through the recursion that  $\text{reg}(r, N) = \log C(r, N)$ ,

$$\begin{aligned} C(1, N) &= 1 \\ C(2, N) &= \sum_{n=0}^N \binom{N}{2} \left(\frac{n}{N}\right)^n \left(\frac{N-n}{N}\right)^{N-n} \\ C(r+2, N) &= C(r+1, N) + \frac{N}{r} C(r, N) \end{aligned} \quad (9.11)$$

as presented in (Kontkanen and Myllymäki, 2007). We note that this can be computed in  $\mathcal{O}(K)$ .

For decomposable models, we propose to implement the qNML by applying the logarithm to the quotient of the one-dimensional NMLs associated to each maximal clique and the NMLs associated with the minimal separators in Eq. (9.1). That is to say, the qNML for decomposable models is given by the expression,

$$s^{qNML}(\mathcal{D}; \mathcal{M}) := \log \frac{\prod_{c \in \mathcal{C}} p_{NML}^1(X_c)}{\prod_{s \in \mathcal{S}} p_{NML}^1(X_s)}. \quad (9.12)$$

where each NML can be assessed according to Eq. (9.10). We can further express the qNML, as in BIC, in terms of a log-likelihood and a penalization term as follows,

$$\begin{aligned} s^{qNML}(\mathcal{D}; \mathcal{M}) &= \log \frac{\prod_{c \in \mathcal{C}} p(X_c; \theta_c^{ML})}{\prod_{s \in \mathcal{S}} p(X_s; \theta_s^{ML})} - \left( \sum_{c \in \mathcal{C}} \text{reg}(\theta_c^{ML}, c) - \sum_{s \in \mathcal{S}} \text{reg}(\theta_s^{ML}, s) \right) \\ &= \log p(\mathcal{D}; \theta^{ML}, \mathcal{M}) - \text{reg}(\theta^{ML}, \mathcal{M}). \end{aligned} \quad (9.13)$$

The qNML score between two decomposable models  $\mathcal{M}^*, \mathcal{M}^c$  is given by

$$s^{qNML}(\mathcal{M}^* \text{ v.s. } \mathcal{M}^c) = \log \frac{p(\mathcal{D}; \theta^{ML}, \mathcal{M}^*)}{p(\mathcal{D}; \theta^{ML}, \mathcal{M}^c)} - (\text{reg}(\theta^{ML}, \mathcal{M}^*) - \text{reg}(\theta^{ML}, \mathcal{M}^c)) \quad (9.14)$$

which can be reduced to the evaluation of the log probabilities and regret differences of the four cliques above  $S_{ab} \cup \{a\}, S_{ab} \cup \{b\}, S_{ab} \cup \{a, b\}, S_{ab}$ , given that both the log-likelihood and the regret terms are also local to the cliques and separators in the chordal graph.

## 9.1.2 Parameter Estimation

Because *Chordalysis* was presented for association discovery, the parameter estimation was simply performed through maximum likelihood. This type of estimation is known to overfit the training data and hence, not appropriate for prediction. In what follows, we describe three techniques to smooth the estimates of the parameters associated with the structure previously uncovered by *Chordalysis*.

### 9.1.2.1 Markov Network m-estimates

The first type of estimate builds a MN (Markov Network) from the uncovered structure and applies m-estimates to JPTs (Joint Probability Tables) (Mitchell, 1997). That is to say, we compute the probability of the data  $x$  given the model  $\mathcal{M}$  as,

$$p(X = x; \theta^{\text{m-est}}, \mathcal{M}) = \frac{\prod_{c \in \mathcal{C}} p(X_c = x_{:c}; \theta^{\text{m-est}})}{\prod_{s \in \mathcal{S}} p(X_s = x_{:s}; \theta^{\text{m-est}})} \quad (9.15)$$

where the joint probability table for a given subset of variables  $\mathcal{A}$ , either a clique or separator, is given by,

$$p(X_{\mathcal{A}} = x_{:\mathcal{A}}; \theta^{\text{m-est}}) = \prod_{a \in \mathcal{A}} \frac{n_{x_a} + m/r_{\mathcal{A}}}{N + m} \quad (9.16)$$



where  $n_{x_a}$  are the observed counts in  $\mathcal{D}$  of variable  $X_a$  taking the value  $x_a$ ,  $m$  is called equivalent sample size and it can be interpreted as if the  $N$  observations were augmented with  $m$  samples distributed evenly over the joint probability table.  $r_{\mathcal{A}}$  is all the possible outcomes in the set  $\mathcal{A}$  defined as  $\prod_{a \in \mathcal{A}} r_a$ , where  $r_a$  is the number of outcomes for variable  $a$ .

### 9.1.2.2 Bayesian Network m-estimates

The second type of estimates builds a BN from the uncovered structure and apply m-estimates to the CPTs (Conditional Probability Tables) (Mitchell, 1997). It is possible to create a BN from the structure because chordal graphs have a PEO (Perfect Elimination Ordering) which can be found through a LBFS (Lexicographic Breadth First Search) with a lexicographic order set as per the decreasing TF-IDF (Term Frequency Inverse Document Frequency) score. Once the order is established, a DAG (Directed Acyclic Graph) can be build and the probabilities estimated by means of m-estimates on the conditional probability tables. That is, the probability of the data  $x$  given the model  $\mathcal{M}$  is,

$$p(X = x; \theta^{\text{m-est}}, \mathcal{M}) = \prod_{v=1 \dots V} p(X_v = x_v | \text{par}(X_v) = x_{\text{par}(X_v)}; \theta^{\text{m-est}}) \quad (9.17)$$

where the conditional probability table for variable  $v$  is computed as follows,

$$p(X_v = x_v | \text{par}(X_v) = x_{\text{par}(X_v)}; \theta^{\text{m-est}}) = \frac{n_{x_v \cup x_{\text{par}(X_v)}} + m/r_v}{N + m} \quad (9.18)$$

where  $n_{x_v \cup x_{\text{par}(x_v)}}$  refers to the observed counts in the collection  $X$  such that  $X_v$  takes the value  $x_v$  and its parents  $\text{par}(X_v)$ , the values  $x_{\text{par}(x_v)}$ .  $m$  is the equivalent sample size and it can be interpreted as if the augmentation was done with  $m$  samples distributed evenly over a row of the conditional probability table. Finally,  $r_v$  is the cardinality of the domain of variable  $v$ .

### 9.1.2.3 Bayesian Network Back-off estimates

We consider a third type of estimate that also builds on the BN structure from the previous section. Furthermore, this estimator orders the variables in the CPT such that those combinations with few data points can use the estimates from their parents, which would embrace more data. This scheme was first introduced for Bayesian classifiers (Friedman et al., 1997) and we refer to it as a Back-off scheme. We expect this technique to improve previous estimators specially in situations with big cliques and/or little data.

For each CPT, we first build a tree with as many levels as parents or conditioning variables. The tree order is also determined through the same TF-IDF ordering established to build the BN. Thus, parents with higher TF-IDF are placed in a higher position in the tree. A node in level  $l$  contains the counts for the value of the conditioned variable  $x_v$  and its parents until that level the  $\text{par}^l(X_v)$ . Therefore, the leaf nodes contain the counts for the conditioned variable with all its parents as in the CPT.

In each node, we first smooth the corresponding probabilities by applying the same m-estimates from Eq. (9.18), but with the set of parents specified by the tree level  $l$ . Then, we

estimate the probability of  $X_v$  given its  $l$  parents as a weighted average of the probability of  $X_v$  from parents at level  $l$  and that at level  $l - 1$ . Mathematically, we can write,

$$p(X_v | \text{par}^l(X_v)) = w \cdot p(X_v | \text{par}^l(X_v)) + (1 - w) \cdot p(X_v | \text{par}^{(l-1)}(X_v)) \quad (9.19)$$

where  $w$  is the weight given by,

$$w = \frac{Np(\text{par}^l(X_v))}{Np(\text{par}^l(X_v)) + N_o} \quad (9.20)$$

where  $p(\text{par}^l(X_v))$  is the marginal probability of variables at level  $l$  and  $N_o$  is the confidence associated with the probability estimate at the previous level. In other words, when  $N_o \gg Np(\text{par}^l(X_v))$  the probability calculated at the previous level of the tree dominates. We also note that if  $N_o = 0$ , node probabilities are calculated only with the information at their level and hence, the probabilities at the leaf nodes are equivalent to that of the BN m-estimates from the previous section.

## 9.2 Related Work

In this section, we present the related models suitable for comparison with the chordal graphical models learned with *Chordalysis*. Fig. 9.1 shows the graphical representation of the variables for the models discussed below for a single document with six words “a”, “b”, “c”, “d”, “e” and “f”. While Bernoulli-Poisson factorisation (blue) uses the local latent variable  $l_n$  to induce correlations among the observed variables, chordal (green) and latent tree (red) models do not consider any local latent variables. However, we note that the latent trees employ the global latent variable  $\mathbf{z} = \{z_1, z_2, z_3\}$  to model the correlations that chordal models can directly induce among the observed variables.

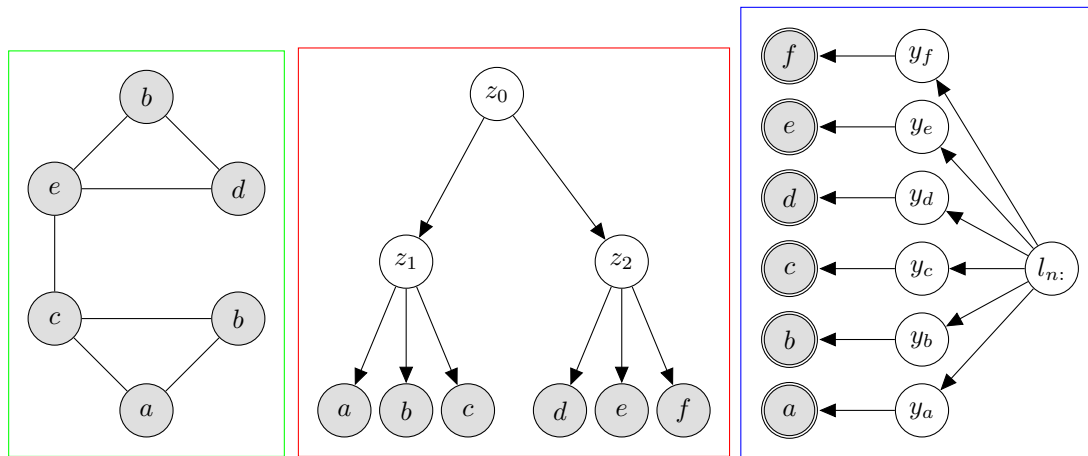


Figure 9.1: Examples of the three graphical models for a single document. Chordal graphical models (green), hierarchical latent tree analysis (red) and Bernoulli-Poisson factorisation (blue).



### 9.2.1 Hierarchical Latent Tree Analysis (HLTA)

HLTA (Liu et al., 2014; Chen et al., 2017) has recently been developed to build hierarchies of topics, where each is defined from the interaction with a limited set of words or other topics. This is achieved by introducing a hierarchy of Boolean latent variables, so that the final model is a tree with the observed words, represented as present/absent, at the leaves. HLTA is comparable to Boolean matrix factorisation, and has been scaled to work with thousands of Boolean variables. The hierarchical nature of the latent variables leads to semantically insightful structures that seem inherently more interpretable than standard topic models (e.g. Chen et al., 2017, Figure 8).

The algorithm to learn these trees from data operates as follows. First, leaves are grouped using a “common latent Boolean factor” statistical test, latent Boolean factors are added and then the grouping process repeated, with progressive expectation-maximization runs to re-estimate probability tables as the trees are grown. This can be done by using all the data (i.e. Batch) or with less accurate mini-batch updating of parameters (i.e. Step-wise) on large data sets.

In (Chen et al., 2017), HLTA was compared with nHDP (nested Hierarchical Dirichlet Process), which is a hierarchical topic model that uses the sequenced representation of text. This comparison has the disadvantage that nHDP is not run natively: it is being trained on Boolean data for which it was not designed. Moreover, the nHDP algorithm used has only demonstrated a marginal improvement in perplexity (Paisley et al., 2015) over HDP-LDA (Teh et al., 2006a). Therefore, we will not compare with nHDP but to another topic model more suitable for binary bagged data presented next.

### 9.2.2 Bernoulli Poisson Factor Analysis (BPFA)

Zhou (2015) introduced the Bernoulli-Poisson link technique to extend Poisson factorisation methods to Boolean data. With this technique, we can take the Poisson factorisation model from Section 2.4.6, which is a flexible model, and add the Bernoulli-Poisson link on top to obtain a representative factor analysis method for Boolean vector data. We refer to this as BPFA (Bernoulli PFA).

The BPFA model does matrix factorisation to create matrices  $\Phi$  (the loading matrix) and  $\Theta$  (the factor matrix) with the following probability forms:

$$\begin{aligned} \phi_k &\sim \text{Dirichlet}_V(\beta \vec{1}) & l_{nk} &\sim \text{Gamma}\left(r_k, \frac{p_k}{1-p_k}\right) \\ y_{np} &\sim \text{Poisson}\left(\sum_{k=1}^K \theta_{nk} \phi_{kp}\right) & b_{np} &= 1_{y_{np} \geq 1}, \end{aligned} \quad (9.21)$$

where  $K$  is the number of topics,  $d$  indexes documents,  $V$  is the size of the vocabulary, and  $\beta$ ,  $\mathbf{r}$  and  $\mathbf{p}$  (where  $0 < p_k < 1$ ) are hyper-parameters with their own priors. Note  $\Phi$  is made up of rows  $\phi_k$  which normalise. This parametrisation means that BPFA is comparable to HDP-LDA. More details can be found in (Zhou et al., 2012; Zhou, 2015; Hu et al., 2016) and Section 2.4.6. The observed data is the Boolean matrix  $\mathbf{Y}$  which has a corresponding latent count matrix  $\mathbf{X}$ .

The algorithm to learn the model parameters from data goes as follows. After random initialization, a Gibbs sampler iterates over parameters and latent counts and topic values. The sampler is built on the conditional exponential family structure of the model, and in

some cases using data augmentation to create fast simple sampling. Hyper-parameters are similarly sampled.

## 9.3 Experimental Methodology

We first discuss general evaluation methods. Because of the variety of different algorithms, this turns out to be a challenging, so we discuss the literature and report our conclusions on how evaluations should be done. We then present the software implementations, data sets and parameter setting for the experiments.

### 9.3.1 Evaluation Methods

Perplexity for topic models is a measure of the *predictive log-likelihood* of a held-out document scaled to a “per-word” measure. Because the probability models in this chapter do not consider counts, we believe it is more appropriate not to scale the log-likelihood in the held-out set and simply report log-likelihood. Furthermore, *Chordalysis* and HLTA do not include local latent variables and hence this quantity can be computed exactly, while BPFA requires unbiased estimation methods to approximate it because of the presence of  $l_{n\cdot}$ . Fortunately, we have studied the likelihood estimation problem for PFA (Poisson Factor Analysis) methods in general and BPFA in particular in Part II. In Chapter 7, we have extended state-of-the-art estimation methods for LDA-based topic models (Wallach et al., 2009c) to PFA and we have proposed new variational methods that are superior to the state-of-the-art in Chapter 8. As a result, we will use the upper-bounded variational importance sampling method to provide an unbiased estimate for the held-out likelihood of BPFA.

Another common evaluation method for factorisation methods is link prediction (Zhou, 2015). The idea is to hold out some of the variables (which may be positive or negative), and then evaluate how well their occurrence is predicted from the remainder of the record. For this to be done correctly, the missing link/variable needs to be made temporarily “missing”, and the various models ran to predict its probability of occurrence. This computation is not always done correctly: some researchers simply set the variable equal to zero rather than treating it as missing. We refer to this task as *omni-directional learning*, as the task is to do predictive modelling, but on a random selection of variables, rather than on a single target variable as is done for classification.

A final task we consider is anomaly detection (Chandola et al., 2009), an important task in security and engineering domains for instance. This is a broad area but we consider the problem of point anomaly detection (whether a single data item or document as an anomaly). There are a broad number of techniques in use, and we use ranking by log probability (lower is more likely to be an anomaly) as a straw-man algorithm to compare with. Note that text anomaly detection is more challenging because of the huge number of variables.

Thus we use three different evaluation protocols, briefly described here, but more detail of implementation is given later.

**Log-likelihood:** Simple measure of predictive probability for documents held out from the training set. Because the exact likelihood is intractable to compute for BPFA, we report an unbiased estimate.

**Omni-directional prediction:** for each document, a variable (word) is drawn at random from a candidate set and then a prediction is made for it (is it in or out of the document). We then evaluate the accuracy of these predictions.

**Anomaly detection:** an infrequent subclass of documents are held out from training and then added to a test set and fed to the model. The subclass are presumably the anomalies in the test set, and are supposed to be low probability documents. Log probability gives a base ranking to predict if it is an anomaly, but other derived measures can be used.

### 9.3.2 Implementation

We modified the Java code of Chordalysis<sup>2</sup> to include qNML and BIC. We used the existing Java code of HLTA<sup>3</sup>. We extended the Matlab code for the BPFA in (Hu et al., 2016) with the sampling of model hyper-parameters. The three evaluations were done in Java and Matlab respectively and care was taken to ensure standardisation across implementations.

Note that most of the evaluations for HLTA and Chordalysis are simple to implement because of their convenient structure as simple Bayesian networks. Suppose that the  $n$ -th document is represented by a Boolean vector  $b_{n:}$ , and if entry  $p$ ,  $b_{np}$ , is converted to a missing value, this is denoted as  $\mathbf{b}_{n-p}$ . Then computation of the measures works as follows, given a specific model  $\mathcal{M}$ .

**Log-likelihood:** we compute  $\log p(b_{n:}|\mathcal{M})$  for CGM (Chordal Graphical Model) and HLTA and approximate it for BPFA. For CGM, this log-likelihood can be computed exactly by summing the logarithms of the entries in the CPTs or JPTs that correspond to the held-out document  $\mathbf{b}$ . For HLTA, we also evaluate the corresponding CPTs entries and sum out the latent variables  $\mathbf{z}$  through exact belief propagation in trees. For BPFA, we approximate this quantity by using the upper bounded variational importance sampler UB-VIS (Upper-Bounded Variational Importance Sampling) presented in Section 8.3.2.

**Omni-directional prediction:** We have a candidate set of words  $S$  and for each  $p \in S$ , we compute  $p(b_{np}|\mathbf{b}_{n-p}, \mathcal{M})$ , and compare it to the correct value  $b_p$  given in the data. That gives a set of  $|S|$  scores calibrated as probabilities, which we measure in terms of AUC (Area Under Curve) by changing the threshold ( $0 < th < 1$ ). We chose AUC-PR (Area Under Curve - Precision Recall) instead of AUC-ROC (Area Under Curve - Receiver Operating Curve) because of the data set skewness (Davis and Goadrich, 2006), i.e there are more absent than present words. Alternatively, we compute root mean square error by averaging  $(b_p - p(b_{np}|\mathbf{b}_{n-p}, \mathcal{M}))^2$  across all words  $S$  and all test documents, and then reporting the square root.

**Anomaly detection:** We test two scores, one based on  $\log p(b_{n:}|\mathcal{M})$  and another, on  $\log p(b_{np}|\mathbf{b}_{n-p}, \mathcal{M})$  for  $p \in S$  and  $S$  consisting of all present words. For both, we train the models holding out the anomalous subclass and we then compute/approximate these scores in a test set that contains anomalous and common documents. As in

<sup>2</sup><https://github.com/fpetitjean/Chordalysis>

<sup>3</sup><https://github.com/kmpoon/hlta>

Omni-directional prediction, we compare these scores for the held-out documents with the true label in terms of AUC-PR. In particular, we show that by adding the term frequencies in the second score, we can achieve higher AUC-PR.

BPFA is the most computationally costly algorithm because it needs to estimate the marginal document likelihood  $\log p(b_{n:}|\mathcal{M})$  and the predictive probabilities  $\log p(b_{np}|\mathbf{b}_{n-p}, \mathcal{M})$ . The former is done through the estimation method discussed earlier. The latter, with a Gibbs sampler that draws samples from the posterior  $p(b_{np}|\mathbf{b}_{n-p}, \mathcal{M})$  through the extended representation given in Algorithm 8.4 together with  $b_{np} \sim \text{Ber}(1 - e^{-\sum_{k=1}^K \phi_{kp} l_{nk}})$ . The estimate for each missing word is then built as,

$$p(b_{np} = 1|\mathbf{b}_{n-p}, \mathcal{M}) \approx \left(1 - e^{-\sum_{k=1}^K \phi_{kp} \bar{l}_{nk}}\right) \quad (9.22)$$

where  $\bar{l}_{n:}$  is an average over the posterior samples.

### 9.3.3 Data sets

We selected three regular and three short text corpora for the experimentation. Collections were preprocessed with the text mining tool assembled in Scala included in the HLTA software<sup>4</sup>, which is suited to build Boolean vector data. For each collection, we tokenised text strings by space, lower-cased tokens, normalised them according to the Normalisation Form KC (NFKC), removed stopwords based on the Lewis list and filtered out words with less than 3 characters. From the resulting vocabularies, we selected the top-500 and top-2000 words with highest TF-IDF score (the raw counts of a term normalised by the negative logarithm of the fraction of documents that contain that term) to build two vocabularies for each collection. All data sets were tokenised and binarised based on these vocabularies and documents without any word were removed. The final Boolean data sets have the following features:

**Twitter:** is extracted in 2011 and 2012 microblog tracks at Text REtrieval Conference (TREC) 3, preprocessed by (Yin and Wang, 2014). It has 11,109 tweets in total and a tweet contains 21 words on average.

**WS:** Web Snippet, used by (Li et al., 2016), contains 12,327 web search snippets and each snippet belongs to one of 8 categories. Documents are typically 15 words long before reducing the vocabulary.

**TMN:** Tag My News, consists of 32,573 English RSS news snippets from Tag My News, used by (Nguyen et al., 2015). Belonging to one of 7 categories, each snippet contains a title and a short description, average length 18 words.

**NIPS:** consists of 1,740 conference papers published at NIPS between 1988 and 1999<sup>5</sup>.

**20NG:** 20Newsgroups, consists of 18,828 news articles and each article is in one of 20 categories<sup>6</sup>. An article has on average 65 different words.

**NYT:** New York Times Annotated Corpus supplied by the Linguistic Data Consortium<sup>7</sup>.

<sup>4</sup><https://github.com/kmpoon/hlta>

<sup>5</sup><http://www.cs.nyu.edu/~roweis/data.html>

<sup>6</sup><http://qwone.com/~jason/20Newsgroups>

<sup>7</sup><http://catalog.ldc.upenn.edu/LDC2008T19>

It contains 1,855,658 news articles. An article has on average 196 different words.

In the likelihood and omni-directional prediction experiments, we looked at the performance of the different graphical models as function of the amount of training data. For each data set, we randomly generated four training splits of different size and evaluated the trained models in a held-out set, which we kept the same for all training splits.

In the omni-directional prediction task, we held-out some words from the test set. In particular, we have randomly selected  $|S| = 10$  words per test document for all data sets except for NIPS, in which we chose  $|S| = 50$  given that the test set was smaller and there was too much variance in the results.

In the anomaly detection task, we used the 20Newsgroups and the WS data set given that both are labeled and they are a good representatives of long and short text. For this task, data sets were split in the classical 80% training, 20% testing framework and the anomalous class was held-out from the training set. To do a fair comparison, we report results by holding out each category in the collection.

In all tasks, each experiment was performed 5 times and different training-test splits were randomly generated at each repetition.

### 9.3.4 Model parameters

Next, we report all model parameters set in this experimentation. A detailed summary can be found in Table 9.1.

For the *Chordalysis* models based on BIC and qNML, we set a safety parameter limiting the tree-width of the network  $K_{max}$  equal to 20 to avoid models with extremely big cliques. Nonetheless, this value was never reached and most of the graphs have smaller tree-width. For Chordalysis with the SMT score, we specified the maximum family-wise error rate  $p_{err} = 0.05$ . For all three *Chordalysis* models, we use the simple Back-Off estimates introduced for Bayesian classifiers (Friedman et al., 1997) that computes each cell in the conditional probability table as a weighted average between the cell itself and its parents, in the probability tree.  $N_0$  controls how much we back-off to the parent estimate, being 0, no back-off and  $\infty$  complete back off to the parent value. As in (Friedman et al., 1997), we use  $N_0 = 5$ . Moreover, we also smooth each cell in the CPT with m-estimates with parameter  $m = 0.5$ .

For HLTA, we used the default values reported by (Chen et al., 2017), except for the structural-batch-size parameter, which we set to 1,000 in the small data sets (20Newsgroups, WS, Twitter) and to 10,000 in the large ones (TMN, NYT).

For BPFA, all hyper-parameters were sampled using benign priors using standard augmented Gibbs samplers. Details of the priors are in Table 9.1.

## 9.4 Experiment Results

In this section, we report the comparison on the experiments presented earlier. For clarity's sake, we first present the assessment of *Chordalysis* models and then we select the best performing model to show the comparison with HLTA and BPFA.

Table 9.1: Model hyper-parameters and algorithm parameters.

| Chord - BIC                            | Chord - qNML                        | Chord - SMT          |
|--|-------------------------------------|----------------------|
| $K_{max} = 20$                         | $K_{max} = 20$                      | $p = 0.05$           |
| $N_o = 5$                              | $N_o = 5$                           | $N_o = 5$            |
| $m = 0.5$                              | $m = 0.5$                           | $m = 0.5$            |
|  |                                     | $p_{err} = 0.05$     |
| BPFA                                   | HLTA - batch                        | HLTA - step          |
| $\beta \sim \text{Ga}(1, 1)$           | max-EM-steps: 50                    | max-EM-steps: 50     |
| $p_k \sim \text{Beta}(1/K, (1 - 1/K))$ | num-EM-starts: 5                    | num-EM-starts: 5     |
| $r_k \sim \text{Ga}(1, 1)$             | EM-threshold: 0.01                  | EM-threshold: 0.01   |
| $K = 200$                              | UD-test-threshold: 3                | UD-test-threshold: 3 |
| train-burnin: 500                      | max-island: 10                      | max-island: 10       |
| train-collect: 500                     | max-top: 15                         | max-top: 15          |
| test-burnin: 100                       | Global-batch-size: 1,000            |                      |
| test-collect: 100                      | Global-max-epochs: 10               |                      |
|  | Global-max-EM-steps: 128            |                      |
|  | Structural-batch-size: 1,000/10,000 |                      |

### 9.4.1 Log-likelihood Experiments

To compare models in terms of held-out log-likelihood, we first train them in one of the four training splits and then compute/approximate the marginal document log-likelihood in the test set. We use the same held-out set for all four train/test splits and repeat the experiment 5 times with different splits. We finally plot the averaged log-likelihood across held-out documents and show the variability across experiments through error bars.

In the first experiment, we compare the likelihood performance of the three *Chordalysis* scores discussed in this chapter: SMT, BIC and qNML. In Fig. 9.2, we plot the averaged log-likelihood of the held-out set (top) and the number of parameters (bottom) as a function of the training documents. As expected, we observe that the proposed criteria for prediction, BIC and qNML, achieve higher log-likelihood than the SMT criterion proposed for association discovery. As shown by the plots at the bottom, this improvement is achieved by learning models with more parameters, i.e. edges in the graph. We also note that qNML results are slightly superior to those of BIC, hence we select this score as a reference for comparison with other models. Note that these experiments were performed with the Back-off method presented in Section 9.1.2.3.

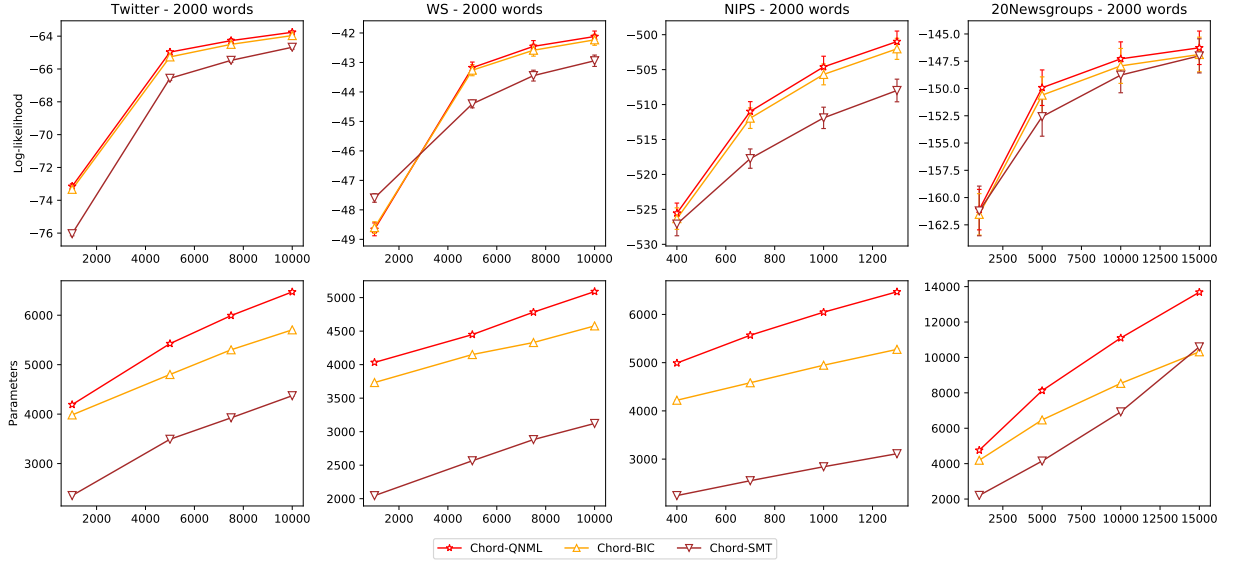


Figure 9.2: Averaged log-likelihood (top) and number of parameters (bottom) as function of training size. Parameters were estimated with the BN Back-off method.

We next compare the three parameter estimations methods presented earlier: MN m-estimates, BN m-estimates and BN Back-off. The top row in Fig. 9.3 displays the averaged log-likelihood as a function of training documents and shows marginally higher log-likelihood for BN m-estimates and Back-off estimators, specially for small training sizes. For this experiment, we chose the chordal graphs learned with the qNML score. Moreover, the bottom row in Fig. 9.3 plots the averaged log-likelihood as a function of the clique size and we can see that the performance of BN Back-off estimates do not drop as fast as the other two methods for large clique sizes. For this experiment, we selected chordal graphs that are saturated up to a clique size. As a consequence of these results, we suggest to use chordal graphs with Back-off parameter estimation for prediction tasks.



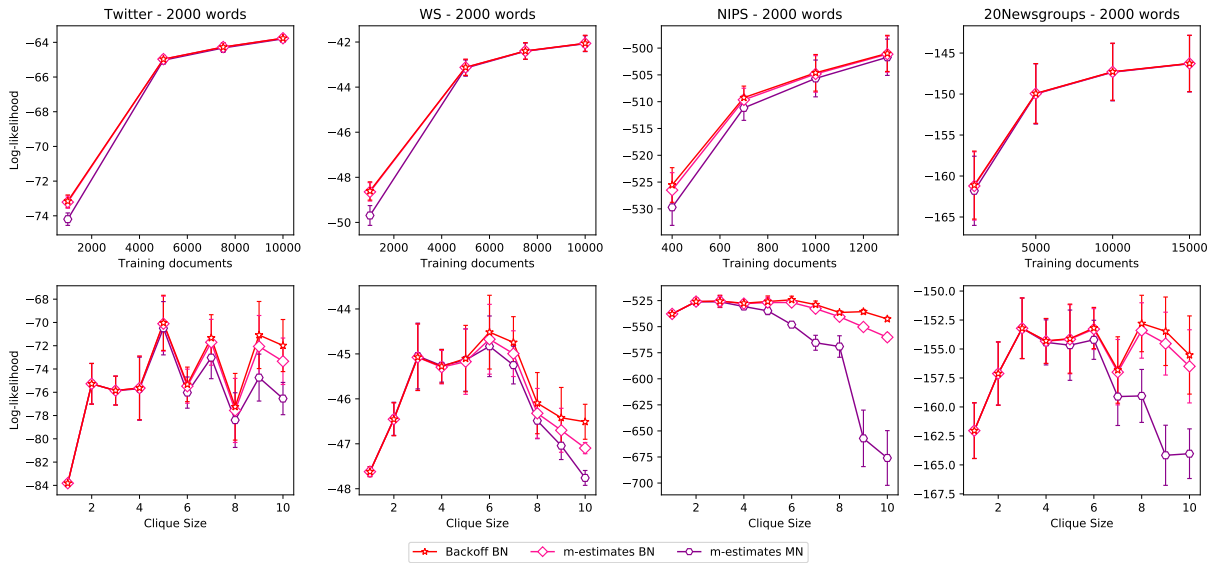


Figure 9.3: Averaged log-likelihood as function of training data (top) and of clique size (bottom). Chordal graphs learned with the qNML score.

Finally, we report in Fig. 9.4 the performance of the two methods discussed in the related work (i.e. HLTA and BPFA) and compare them to *Chordalysis* with qNML metric. Note that we next only provide results for the batch version of HLTA (HLTA - batch) since its stepwise version (HLTA - step) has always worse performance for these collections. As it is shown in the plots, chordal models are superior to hierarchical latent trees and Poisson factorisation in short text collections, whereas Poisson factorisation is superior in regular text collections. *Chordalysis*' performance is comparable to that of HLTA in the large training set regime in NIPS and 20Newsgroups. We also observe that the performance of chordal models improves more than its competitors with the training size, so it would be interesting to see these results for bigger data sets.

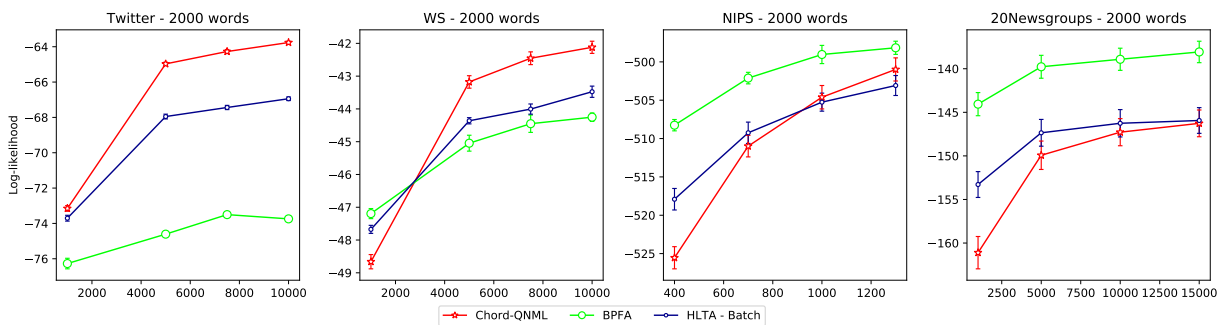


Figure 9.4: Averaged log-likelihood as a function of training size.

In all experiments above, we have found almost no difference between the log-likelihood result in document collections with 500 and 2000 words vocabulary, so we only reported the later. We next report results for the biggest document collections (i.e. TMN and NYT) with a vocabulary of 500 words for those methods that are able to scale to these dimensions, as depicted in Fig. 9.5. More concretely, these methods are the HLTA - step, BPFA and



chordal models for TMN and the HLTA - step and chordal models for NYT. We observe that *Chordalysis* with qNML achieves the higher log-likelihood in short and regular data sets.

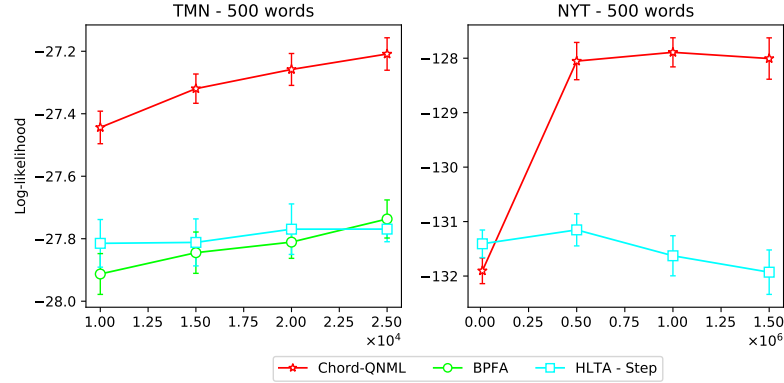


Figure 9.5: Averaged log-likelihood as a function of training size in big collections.

### 9.4.2 Omni-directional Prediction Experiments

For each document in the test set we randomly hold out a set of  $S$  words ( $|S| = 50$  in NIPS,  $|S| = 10$  in the rest) and predict their presence or absence given all other words in the document and the model. Predictions are compared to the true word labels (present or absent) and assessed in terms of AUC-PR and RMSE (Root Mean Squared Error). Higher is better for AUC-PR, whereas lower is better for RMSE.

Fig. 9.6 shows the AUC-PR (top) and RMSE (bottom) for the different scoring functions of *Chordalysis* as a function of the number of training documents. Similar to the log-likelihood task, qNML achieves higher AUC-PR across all data sets and training set ranges followed by BIC, despite SMT achieves lower RMSE in small training sets. We select *Chordalysis* with qNML score for the remaining experiments.

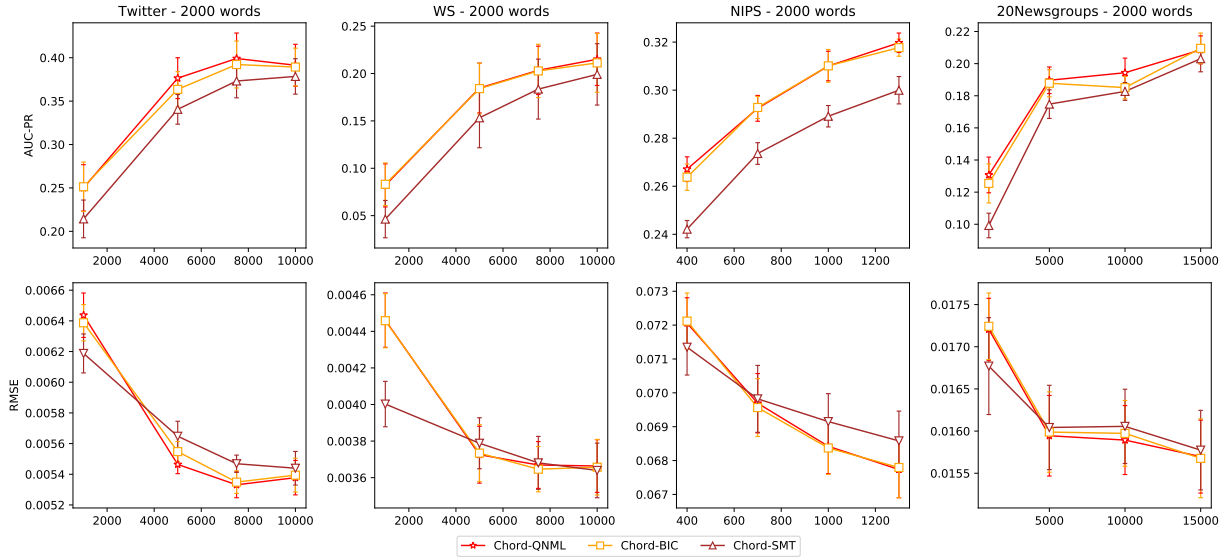


Figure 9.6: AUC-PR scores (top) and RMSE (bottom) as a function of training documents for Chordal graphs learned with Back-off estimates.

Fig. 9.7 shows the comparison of the related method in the omni-directional prediction task. As depicted, *Chordalysis* achieves highest AUC-PR in short text collections as well as in NIPS, whereas BPFA is superior in 20Newsgroups. We also observe that RMSE results are not completely correlated with those of AUC-PR in small training regimes.

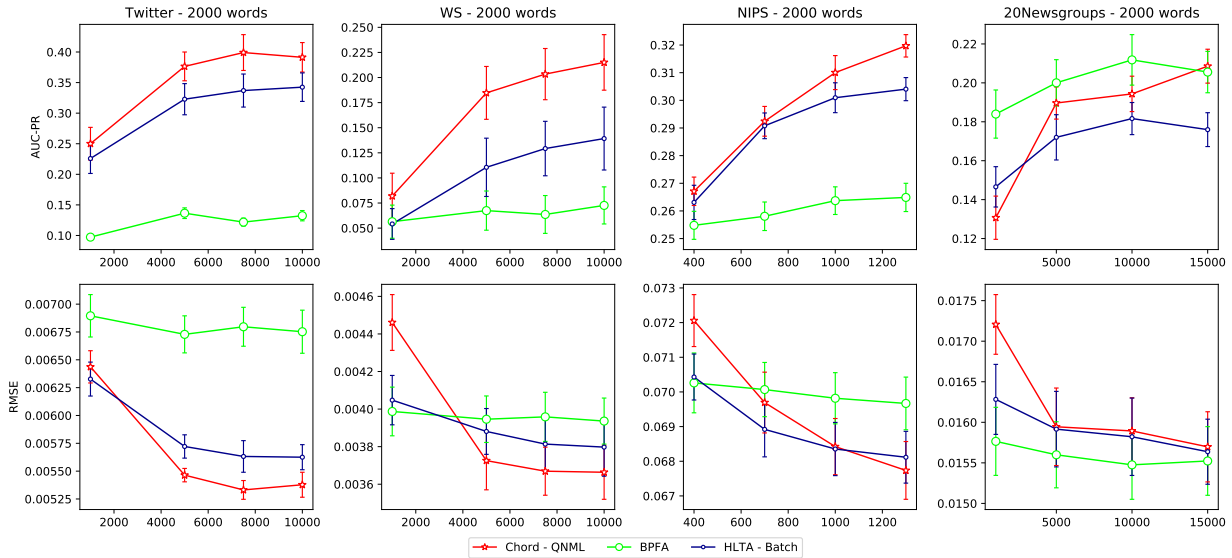


Figure 9.7: AUC-PR scores (top) and RMSE (bottom) as a function of training documents.

These results confirm that the *Chordalysis* and HLTA have higher prediction power in short text, whereas BPFA is superior in regular text collections. Since neither *Chordalysis* nor HLTA have local latent variables, we validate our hypothesis that models without local latent variables should achieve better prediction in short text.

### 9.4.3 Anomaly Detection Experiments

A statistical approach to assess how anomalous a test document consist sin computing its log-likelihood under a model which has been learned from the normal or typical documents, i.e. the null model (Chandola et al., 2009). If the log-likelihood is low, the document cannot be explained well by the null model and hence, it is likely to be an anomalous document. Next, we compare the different graphical models discussed earlier on this new task. To do this, we first train each model in a collection without documents from the anomalous class and we then compute the log-likelihood for each of the test documents, which might be either from the anomalous or the normal class.

In Table 9.2, we show the AUC-PR figures obtained from considering the anomalous class for each of the WS categories. For each category, we trained all models on the training split of 10,000 documents, excluded documents from the anomalous class and evaluated the log-likelihood of test documents. We observe that Chordalysis with qNML obtains the highest AUC-PR figures across all categories and HLTA - batch the lowest in 5 categories.

Table 9.2: AUC-PR figures for the log-likelihood anomaly score in the WS collection. The highest figure is **boldfaced** and the lowest, underlined.

| Category                   | Chord - qNML                      | BPFA                              | HLTA - batch                      |
|----------------------------|-----------------------------------|-----------------------------------|-----------------------------------|
| business                   | <b>0.9272</b> $\pm$ <b>0.0054</b> | 0.923 $\pm$ 0.0055                | <u>0.9222</u> $\pm$ <u>0.0055</u> |
| computers                  | <b>0.9138</b> $\pm$ <b>0.0052</b> | <u>0.9069</u> $\pm$ <u>0.0055</u> | 0.9099 $\pm$ 0.0053               |
| culture-arts-entertainment | <b>0.8343</b> $\pm$ <b>0.0029</b> | 0.8272 $\pm$ 0.003                | <u>0.8269</u> $\pm$ <u>0.0029</u> |
| education-science          | <b>0.8518</b> $\pm$ <b>0.003</b>  | <u>0.8412</u> $\pm$ <u>0.0044</u> | 0.8436 $\pm$ 0.0035               |
| engineering                | <b>0.9707</b> $\pm$ <b>0.0032</b> | <u>0.9677</u> $\pm$ <u>0.0028</u> | 0.9693 $\pm$ 0.003                |
| health                     | <b>0.9429</b> $\pm$ <b>0.0036</b> | 0.9408 $\pm$ 0.0034               | <u>0.9382</u> $\pm$ <u>0.0032</u> |
| politics-society           | <b>0.9072</b> $\pm$ <b>0.0023</b> | 0.9024 $\pm$ 0.0021               | <u>0.9016</u> $\pm$ <u>0.0019</u> |
| sports                     | <b>0.9331</b> $\pm$ <b>0.0028</b> | 0.9303 $\pm$ 0.0027               | <u>0.9284</u> $\pm$ <u>0.0027</u> |

Table 9.3 presents the results for the 20Newsgroups data set trained in the data split of 15,000 documents and excluding the anomalous class. In this scenario, BPFA achieves the highest score across more categories than Chordalysis qNML, but this obtains less lowest AUC-PR scores than the rest, meaning that it has an average performance . These results confirms that Chordalyis qNML is particularly good for prediction tasks in short text collections, whereas BPFA is superior in regular collections.

Table 9.3: AUC-PR figures for the log-likelihood anomaly score in the 20Newsgroups collection. The highest figure is **boldfaced** and the lowest, underlined.

| Category               | Chord - qNML                      | BPFA                              | HLTA - batch                      |
|------------------------|-----------------------------------|-----------------------------------|-----------------------------------|
| sci_electronics        | <u>0.9378</u> $\pm$ 0.0032        | <b>0.9392</b> $\pm$ <b>0.003</b>  | 0.9382 $\pm$ 0.003                |
| sci_space              | 0.9495 $\pm$ 0.0023               | <b>0.9507</b> $\pm$ <b>0.0022</b> | <u>0.949</u> $\pm$ <u>0.0023</u>  |
| comp_sys_ibm           | <u>0.9442</u> $\pm$ 0.0052        | 0.9443 $\pm$ 0.0051               | <b>0.9444</b> $\pm$ <b>0.0052</b> |
| comp_graphics          | 0.9364 $\pm$ 0.0043               | <u>0.9363</u> $\pm$ <u>0.004</u>  | <b>0.9366</b> $\pm$ <b>0.004</b>  |
| rec_motorcycles        | 0.94 $\pm$ 0.0035                 | <b>0.9402</b> $\pm$ <b>0.0034</b> | <u>0.9391</u> $\pm$ 0.0036        |
| sci_crypt              | 0.9715 $\pm$ 0.0013               | <b>0.9722</b> $\pm$ <b>0.0013</b> | <u>0.9713</u> $\pm$ <u>0.0013</u> |
| soc_religion_christian | <b>0.9667</b> $\pm$ <b>0.0017</b> | <u>0.9659</u> $\pm$ <u>0.0019</u> | 0.9662 $\pm$ 0.0017               |
| talk_religion_misc     | <b>0.9749</b> $\pm$ <b>0.002</b>  | <u>0.9746</u> $\pm$ <u>0.002</u>  | 0.9749 $\pm$ 0.002                |
| rec_sport_hockey       | <b>0.9484</b> $\pm$ <b>0.001</b>  | <u>0.9467</u> $\pm$ <u>0.0013</u> | 0.9477 $\pm$ 0.0011               |
| alt_atheism            | <b>0.9712</b> $\pm$ <b>0.001</b>  | 0.971 $\pm$ 0.0009                | <u>0.9709</u> $\pm$ <u>0.001</u>  |
| rec_sport_baseball     | <b>0.9393</b> $\pm$ <b>0.0017</b> | <u>0.9383</u> $\pm$ <u>0.0017</u> | 0.9392 $\pm$ 0.0017               |
| comp_windows_x         | <u>0.9472</u> $\pm$ <u>0.0024</u> | 0.9473 $\pm$ 0.0023               | <b>0.9477</b> $\pm$ <b>0.0024</b> |
| sci_med                | 0.9446 $\pm$ 0.0028               | <b>0.945</b> $\pm$ <b>0.0029</b>  | <u>0.9436</u> $\pm$ <u>0.003</u>  |
| talk_politics_guns     | 0.9699 $\pm$ 0.0018               | <b>0.9705</b> $\pm$ <b>0.0017</b> | <u>0.9695</u> $\pm$ <u>0.0018</u> |
| rec_autos              | 0.9422 $\pm$ 0.0017               | <b>0.9425</b> $\pm$ <b>0.0016</b> | <u>0.9411</u> $\pm$ <u>0.0018</u> |
| talk_politics_mideast  | <b>0.9735</b> $\pm$ <b>0.0019</b> | 0.9734 $\pm$ 0.0018               | <u>0.9732</u> $\pm$ <u>0.0018</u> |
| comp_sys_mac_hardware  | <u>0.9415</u> $\pm$ <u>0.0014</u> | <b>0.9418</b> $\pm$ <b>0.0013</b> | 0.9417 $\pm$ 0.0014               |
| comp_os_ms             | 0.9387 $\pm$ 0.002                | <u>0.9381</u> $\pm$ <u>0.002</u>  | <b>0.9389</b> $\pm$ <b>0.0019</b> |
| talk_politics_misc     | 0.9754 $\pm$ 0.0018               | <b>0.9756</b> $\pm$ <b>0.0018</b> | <u>0.9752</u> $\pm$ <u>0.0018</u> |
| misc_forsale           | <u>0.9277</u> $\pm$ <u>0.0029</u> | 0.9279 $\pm$ 0.0028               | <b>0.931</b> $\pm$ <b>0.0031</b>  |

In what follows, we argue that the log-likelihood of a document given a null model might not be enough to find anomalies in binarised text, because the model disregards that there are words which are more common than others. As a result, a test document that mostly contains common words will obtain a high log-likelihood score even if it is anomalous. Therefore, we propose to instead compute an anomaly score that weights the probability of each word by the IDF (Inverse Document Frequency). For each present word in a document from the test set, we compute the anomaly score  $Sc$  as,

$$Sc = \frac{1}{|S|} \sum_{p \in S} \log p(b_{np} | \mathbf{b}_{n-p}, \mathcal{M}) \log \frac{N}{1 + c_p} \quad (9.23)$$

where  $N$  are the number of documents in the training set,  $c_p$ , the counts of word  $p$  across all training documents and  $S$ , the set of present words in the  $n$ -th test document.

In Table 9.4, we demonstrate that models with this score achieve always higher AUC-PR figures than with the log-likelihood score. The baseline column refers to the best performing model from Table 9.2 which now achieves the lowest score when compared to the models with this new score. Moreover, the HLTA - batch now achieves the best performing model, although we notice that the differences are not significant in most categories. This indicates us that the Inverse Document Frequency term is playing a more important role than the probability of the word in discriminating between anomalous and normal documents.

Table 9.4: AUC-PR figures for the IDF-weighted anomaly score in the WS collection. The highest figure is **boldfaced** and the lowest, underlined.

| Category                   | Baseline                   | Chord - qNML                      | BPFA                             | HLTA - batch                      |
|----------------------------|----------------------------|-----------------------------------|----------------------------------|-----------------------------------|
| business                   | <u>0.9272</u> $\pm$ 0.0054 | <b>0.9656</b> $\pm$ <b>0.0033</b> | 0.9633 $\pm$ 0.0038              | 0.9642 $\pm$ 0.0019               |
| computers                  | <u>0.9138</u> $\pm$ 0.0052 | 0.9562 $\pm$ 0.0021               | 0.9552 $\pm$ 0.0019              | <b>0.9578</b> $\pm$ <b>0.002</b>  |
| culture-arts-entertainment | <u>0.8343</u> $\pm$ 0.0029 | 0.9323 $\pm$ 0.0017               | 0.9312 $\pm$ 0.0019              | <b>0.9325</b> $\pm$ <b>0.0017</b> |
| education-science          | <u>0.8518</u> $\pm$ 0.003  | 0.9152 $\pm$ 0.0041               | 0.9086 $\pm$ 0.0052              | <b>0.9181</b> $\pm$ <b>0.0038</b> |
| engineering                | <u>0.9707</u> $\pm$ 0.0032 | 0.9911 $\pm$ 0.002                | 0.9908 $\pm$ 0.0017              | <b>0.9919</b> $\pm$ <b>0.0019</b> |
| health                     | <u>0.9429</u> $\pm$ 0.0036 | 0.9695 $\pm$ 0.002                | <b>0.972</b> $\pm$ <b>0.0005</b> | 0.969 $\pm$ 0.0015                |
| politics-society           | <u>0.9072</u> $\pm$ 0.0023 | 0.9608 $\pm$ 0.0034               | <b>0.962</b> $\pm$ <b>0.0035</b> | 0.9602 $\pm$ 0.0037               |
| sports                     | <u>0.9331</u> $\pm$ 0.0028 | <b>0.9749</b> $\pm$ <b>0.0031</b> | 0.974 $\pm$ 0.003                | 0.9723 $\pm$ 0.0024               |

Finally, we report the results of the IDF-weighted score for the 20Newsgroups collection. In this data set, we observe that the Baseline, which is the best performing model from Table 9.3, is not always the worst. This can be explained from the fact that the IDF weights are smaller in regular text than in short text because terms tend to occur more frequently across documents. Therefore, the impact of the IDF term in the 20Newsgroups is smaller than in WS and for some categories it can even be harmful. Nonetheless, the Baseline achieves the worst performance for most of the categories and Chordalysis with qNML is superior in more categories than any other model.

Table 9.5: AUC-PR figures for the IDF-weighted anomaly score in the 20Newsgroups collection. The highest figure is **boldfaced** and the lowest, underlined.

| Category               | Baseline                          | Chord - qNML                      | BPFA                              | HLTA - batch                      |
|------------------------|-----------------------------------|-----------------------------------|-----------------------------------|-----------------------------------|
| sci_electronics        | <u>0.9392</u> $\pm$ 0.003         | 0.9634 $\pm$ 0.0011               | <b>0.965</b> $\pm$ <b>0.0014</b>  | 0.9644 $\pm$ 0.0007               |
| sci_space              | <u>0.9507</u> $\pm$ 0.0022        | <b>0.9691</b> $\pm$ <b>0.0013</b> | 0.9647 $\pm$ 0.0014               | 0.9665 $\pm$ 0.0021               |
| comp_sys_ibm           | <u>0.9444</u> $\pm$ 0.0052        | <b>0.9632</b> $\pm$ <b>0.0014</b> | 0.9595 $\pm$ 0.0014               | 0.9585 $\pm$ 0.0022               |
| comp_graphics          | <u>0.9364</u> $\pm$ 0.0043        | <b>0.9639</b> $\pm$ <b>0.002</b>  | 0.9627 $\pm$ 0.001                | 0.9629 $\pm$ 0.0016               |
| rec_motorcycles        | <u>0.94</u> $\pm$ 0.0035          | 0.9758 $\pm$ 0.0009               | <b>0.9789</b> $\pm$ <b>0.0009</b> | 0.9782 $\pm$ 0.001                |
| sci_crypt              | 0.9722 $\pm$ 0.0013               | <b>0.9746</b> $\pm$ <b>0.0009</b> | <u>0.9663</u> $\pm$ <u>0.0015</u> | 0.9691 $\pm$ 0.0014               |
| soc_religion_christian | <b>0.9667</b> $\pm$ <b>0.0017</b> | 0.9422 $\pm$ 0.0026               | 0.9332 $\pm$ 0.0022               | <u>0.9303</u> $\pm$ <u>0.0026</u> |
| talk_religion_misc     | <b>0.9749</b> $\pm$ <b>0.002</b>  | 0.9607 $\pm$ 0.001                | <u>0.9591</u> $\pm$ <u>0.0008</u> | 0.9599 $\pm$ 0.0008               |
| rec_sport_hockey       | <u>0.9484</u> $\pm$ <u>0.001</u>  | <b>0.9706</b> $\pm$ <b>0.0002</b> | 0.9622 $\pm$ 0.002                | 0.9693 $\pm$ 0.0016               |
| alt_atheism            | <b>0.9712</b> $\pm$ <b>0.001</b>  | 0.9597 $\pm$ 0.0015               | <u>0.9553</u> $\pm$ <u>0.0025</u> | 0.9566 $\pm$ 0.0015               |
| rec_sport_baseball     | <u>0.9393</u> $\pm$ <u>0.0017</u> | 0.9585 $\pm$ 0.0017               | 0.955 $\pm$ 0.0017                | <b>0.9627</b> $\pm$ <b>0.0013</b> |
| comp_windows_x         | <u>0.9477</u> $\pm$ <u>0.0024</u> | <b>0.9774</b> $\pm$ <b>0.0015</b> | 0.9711 $\pm$ 0.0026               | 0.9739 $\pm$ 0.0019               |
| sci_med                | <u>0.945</u> $\pm$ <u>0.0029</u>  | 0.9623 $\pm$ 0.0016               | <b>0.9641</b> $\pm$ <b>0.0007</b> | 0.9596 $\pm$ 0.0017               |
| talk_politics_guns     | <b>0.9705</b> $\pm$ <b>0.0017</b> | 0.951 $\pm$ 0.0017                | 0.949 $\pm$ 0.0026                | <u>0.9481</u> $\pm$ <u>0.0022</u> |
| rec_autos              | <u>0.9422</u> $\pm$ <u>0.0017</u> | 0.9619 $\pm$ 0.0023               | 0.9646 $\pm$ 0.0014               | <b>0.9651</b> $\pm$ <b>0.0019</b> |
| talk_politics_mideast  | 0.9735 $\pm$ 0.0019               | <b>0.977</b> $\pm$ <b>0.0009</b>  | <u>0.9647</u> $\pm$ <u>0.0013</u> | 0.9651 $\pm$ 0.0012               |
| comp_sys_mac_hardware  | <u>0.9417</u> $\pm$ <u>0.0014</u> | <b>0.9683</b> $\pm$ <b>0.0012</b> | 0.9675 $\pm$ 0.0007               | 0.9678 $\pm$ 0.0011               |
| comp_os_ms             | <u>0.9389</u> $\pm$ <u>0.0019</u> | <b>0.9626</b> $\pm$ <b>0.0014</b> | 0.9589 $\pm$ 0.0013               | 0.959 $\pm$ 0.002                 |
| talk_politics_misc     | <b>0.9754</b> $\pm$ <b>0.0018</b> | 0.9613 $\pm$ 0.0011               | <u>0.9577</u> $\pm$ <u>0.0018</u> | 0.9586 $\pm$ 0.0017               |
| misc_forsale           | <u>0.931</u> $\pm$ <u>0.0031</u>  | <b>0.9761</b> $\pm$ <b>0.001</b>  | 0.976 $\pm$ 0.0008                | 0.9759 $\pm$ 0.0012               |

#### 9.4.4 Running Times

Finally, we measured the running times for the 6 models under study in the 20Newsgroups collection. Although these running times have been measured in similar conditions, the

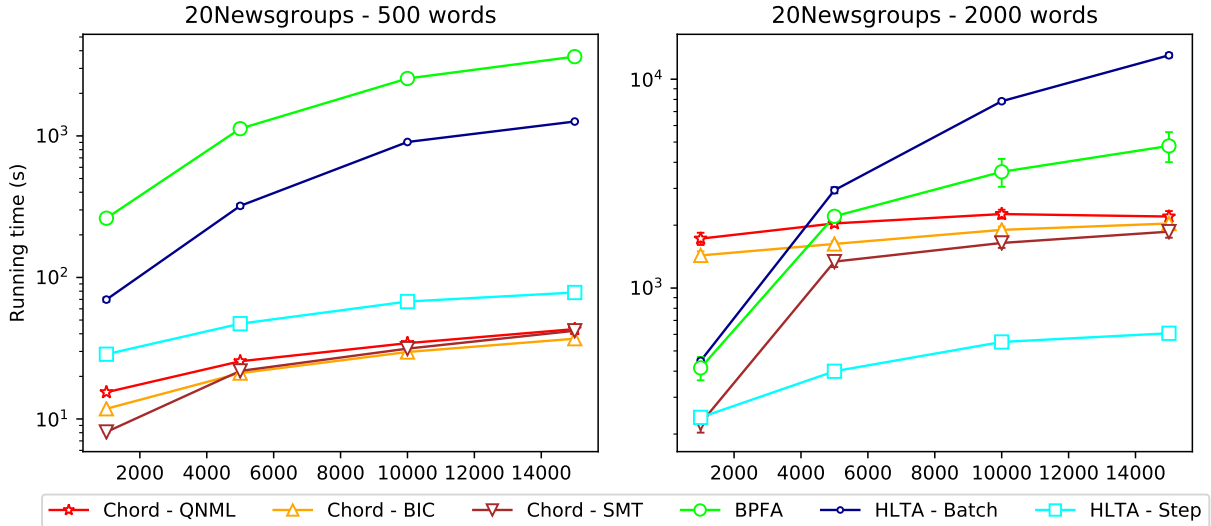


Figure 9.8: Running times as a function of the training documents in the 20Newsgroups collections with a vocabulary of 500 (left) and 2,000 (right) words.

implementations do not only differ from the algorithmic point of view, but also from the programming language used. Therefore, the aim of these results is simply to provide some insights on the scaling performance of each algorithm rather than providing an absolute time comparison.

As we can see in Fig. 9.8, BPFA and HLTA - batch are the most time consuming algorithms, specially for large training sizes. This hampered the execution of these algorithms in large data sets, where the HLTA - step algorithm had to be used. We notice that the running time of HLTA - batch surpasses that of BPFA when changing the vocabulary size from 500 words to 2,000 words. This is due to the fact that BPFA does not depend on the vocabulary size but on the number of present words, whereas the HLTA - batch depends on the vocabulary size.

We also note that the running time of Chordalysis models exceed that of HLTA - step when changing the vocabulary size from 500 to 2,000 words. This hampers the use of Chordalysis models for vocabularies larger than thousands of words.

Although the differences between the three Chordalysis scores are small, we highlight the fact that qNML takes more time than BIC and SMT. This is due to the fact that the search for qNML takes more steps than that of BIC and SMT given that the qNML score is more optimistic.

The HLTA - step algorithm keeps the running time quite steady across different vocabulary sizes and training sets by sub-sampling the vocabulary to learn the structure (structural-batch-size in Table 9.1) and sub-sampling the observations to learn the parameters (Global-batch-size in Table 9.1). However, we found that the performance penalty was often severe for this model.

### 9.4.5 Summary of Results

Finally, we summarise the main findings from the previous experiments:

- *Chordalysis* with the qNML scoring function achieved a higher held-out likelihood across collections than *Chordalysis* with BIC or SMT.
- While the three parameter estimation methods performed similarly in small cliques, Back-off estimates performed better in models with bigger cliques. Therefore, we proposed to use this method together with the qNML score for learning graphical models in binarised text collections.
- *Chordalysis* with qNML performed better than existing topic models (HLTA and BPFA) in short text data sets (Twitter & WS), whereas BPFA did better in long text collections (NIPS & 20Newsgroups).
- Similar conclusions were drawn from the omni-directional prediction and the anomaly tasks, in which *Chordalysis* with qNML tended to perform better in short text and BPFA and HLTA in longer text.
- Experiments also showed that the running time of *Chordalysis* models grows slower than BPFA and HLTA - Batch models with the number of training documents, but faster than HLTA - stepwise. The held-out likelihood of HLTA - stepwise is much lower than *Chordalysis* in big document collections (TNM & NYT).

## 9.5 Conclusion

In this chapter, we presented an algorithm, known as *Chordalysis*, for learning chordal graphical models on binarised text. However, the scoring metrics used by the original algorithm were thought for association discovery and hence, not appropriate for prediction tasks. Therefore, we introduced new scoring metrics and parameter estimation techniques more suitable for the tasks at hand. Through experimentation in different text collections, we demonstrated that the qNML score and Back-off estimates performed better in short and regular text than the early proposed methods (SMT and maximum likelihood estimates). Therefore, we selected qNML and Back-off estimates for comparing *Chordalysis* with the rest of models.

We then sought to compare three very different styles of unsupervised text models that are based on the bagged and binarised representation of text: *Chordalysis*, which learns chordal graphs, HLTA, which learns trees with observed variables at the leaves, and BPFA, which is a matrix factorisation method. For evaluation we used document log-likelihood, omni-directional prediction, and anomaly detection. Document log-likelihood was computed exactly for HLTA and *Chordalysis* models thanks to their tree and chordal structures and it was approximated for BPFA models through the unbiased estimators presented in Part II, in particular the upper-bounded variational importance sampler. The omni-directional prediction task could be performed again in an exact manner for trees and chordal models and it was estimated via a Gibbs sampling scheme for BPFA. Finally, the anomaly detection reused these two calculations to build two distinct anomaly scores.

For document log-likelihood, *Chordalysis* was superior in short text (Twitter & WS) and BPFA, in regular/long text (NIPS & 20Newsgroups). The performance of HLTA was also superior to that of BPFA in short text, confirming that the lack of local latent variables is helpful in this situation. Moreover, we showed that *Chordalysis* scales to big document

collections and its performance improves with larger training sets, surpassing HLTA - step in TMN and NYT collections and BPFA in the TMN collection.

For the omni-directional prediction task, we reported similar results to the document log-likelihood task, except for the NIPS data set, where Chordalysis was superior to BPFA. We also noted that the RMSE results were not correlated with AUC-PR in the small training set regime.

For anomaly detection with the log-likelihood score, Chordalysis was superior in short text (i.e. WS) and BPFA, in regular/long text (i.e. 20Newsgroups). We showed that the IDF-weighted score was more discriminative than the log-likelihood score, specially in short text, encouraging the use of word count information for anomaly detection in text.

We also showed that whereas Chordalysis scaled well with the number of training documents, it did not scale with the vocabulary size. BPFA, whose time complexity does not depend on the vocabulary size but on the number of non-zero words, scaled much better with the vocabulary than with the training set. Finally, HLTA - batch had troubles to scale with both the vocabulary and training set, and the HLTA - step, which scales in both, suffered a severe performance penalty.



# 10

## Future Work and Conclusion

*“Ancora Imparo”*

Unknown

To close this dissertation, we first present open problems and future research derived from this work and then, we summarise the main results and conclusions.

The future work is presented per part since each has finally resulted in a separate research line. We expect that the independent progresses in each part can contribute to move forward the whole field of probabilistic topic models, for both short and long text.

The conclusion reviews the main points addressed by this thesis and it is structured to answer the research questions posed in the introduction.

### 10.1 Future Work

#### 10.1.1 Part I: Event Detection

The concept of event was first discussed in the context of news media as some unique thing happening at some point in time. The community then focused on building detection methods that were capable of uncovering these event from media stories. Similarly, an event in the context of social networks was defined as something that causes a large number of actions in the network, and we then built detection techniques to identify these increases. In our opinion, the descriptors “some unique thing” or “a large number of” are often too vague to enable the identification, even manually, of events and hence, we need to encourage more precise definitions of what constitutes an event. Furthermore, the fact that events cannot be measured directly, but we can only measure their effects, i.e. media stories or social network actions, makes the problem of event detection even more challenging. Therefore, we believe that the task requires to work hand in hand with domain experts who perfectly know the characteristic of the events that have to be detected. By eliciting knowledge from them,

not only can we identify which events can be measured from their effects, but also be more precise about their formal definitions. In fact, we think that the definition of event varies across domains, and hence the detection techniques to uncover them need to be flexible to adapt as well.

In this thesis, we have addressed the problem of event detection in an unsupervised manner, in which we aimed to identify these latent or hidden events that best explained the tweets that we observed. However, when building and tuning up the techniques, we brought expert knowledge about the events, such as events in “La Mercè” did not cover the whole city of Barcelona or they did not span during the whole week. Furthermore, when evaluating the discovered events, we had a ground truth data set built from our own expert knowledge of the festivities. Nonetheless, we believe it is necessary to elicit this knowledge from a final user of the application. For example, the city authorities of Barcelona could be a better final user for “La Mercè” test bed.

Along this line, the progress of this research field is also tied to the creation and public release of this expert knowledge in the form of data sets or services to evaluate the results. In the case of Twitter, the publication of tweets’ IDs, as done in this thesis or in (McMinn et al., 2013), seems an interesting way to go, although the deletion or privatization of tweets by their owners can prevent the full reproducibility of research. In a similar way, evaluation services, like the one adopted by the TREC (Text REtrieval Conference) in which researchers submit their results to a service that returns the evaluation score, could also be employed.

Regarding the detection methods proposed in this thesis, we have seen that fully probabilistic approaches enable the inclusion of domain knowledge in a principled way. Furthermore, the learning of the probability model can also be formulated in an integral manner, avoiding to train things separately like in Tweet-SCAN. However, the WARBLE model contained many more hyper-parameters than Tweet-SCAN and some of them are quite critical, such as the number of events and topics. Therefore, future work should consider non-parametric extensions of WARBLE to infer these critical hyper-parameters, for instance, through HDP (Hierarchical Dirichlet Process) (Teh et al., 2006b). Similarly, non-parametric density estimation could be used to integrate the learning of backgrounds into the same scheme.

### 10.1.2 Part II: Likelihood Evaluation

To the best of our knowledge, the estimation of the marginal document likelihood for PFA (Poisson Factor Analysis) had not been considered until this thesis. However, we have shown that its unbiased estimation is a prerequisite for comparing the prediction performance of PFA with other models that work on bagged counts. Therefore, we believe that this research line can advance in different ways.

First, there exist many estimation methods developed for LDA (Latent Dirichlet Allocation) and other LVMs (Latent Variable Models). Although their extension is not always trivial for PFA, it would be interesting to look more in depth at methods like the AIS (Annealed Importance Sampling) (Neal, 2001) or the Chib-style estimator (Murray and Salakhutdinov, 2009), as we did for the Left-to-right sequential sampler.

Second, the VIS (Variational Importance Sampling) presented in Chapter 8 used the KL (Kullback-Leibler) divergence to approximate the optimal proposal with a mean-field distri-

bution. Future work could consider other divergence metrics, such as the  $\chi$ -divergence (Ding et al., 2017), or other variational approximations such as inference networks (Kingma and Welling, 2014).

Third, the variational bounds developed in Chapter 8 to sandwich the marginal document likelihood and determine the accuracy of these estimators in realistic scenarios were often too loose. Therefore, we need to develop tighter bounds to this marginal in order to discriminate between different methods in realistic setups. The stochastic bounds proposed in (Grosse et al., 2015) could be particularized for PFA.

Last, exploring the use of the proposed estimation methods as well as the exact calculation for other evaluation tasks like document completion or word prediction is another interesting avenue for future work. These tasks have been performed through specialized samplers (Zhou et al., 2012) but rigorous studies to evaluate how well they correlate with the exact computation have not yet been conducted.

On another front, we showed that the upper-bounded VIS method built from the minimisation of the forward KL achieved state-of-the-art accuracy and convergence results for different PFA models. As far as we know, the use of the forward KL has not yet considered for learning PFA models, but it could potentially learn models with greater prediction power than the reverse KL. Similarly, the use of the upper-bounded proposal for likelihood estimation in LDA has not yet been explored, but it could also improve the accuracy of the mean-field importance sampling methods which are based on the lower bound approximation (Buntine, 2009).

### 10.1.3 Part III: Chordal models

In the last part of this thesis, we showed how to build chordal models for binarised text through an algorithm called *Chordalysis* that scales to a vocabulary of thousands of words. Nonetheless, most topic models scale with the number of non-zero words in the corpora, which is much lower than the vocabulary size ( $\bar{V}_c \ll V$ ), and hence *Chordalysis* cannot compete with them in ever-increasing vocabularies. Future work should address the scaling to larger vocabularies or completely remove its dependency with the size.

Another important venue for future work consists in extending the CGMs (Chordal Graphical Models) to count data. This would suppose a major re-definition of the *Chordalysis* algorithm which was thought for categorical data, and hence not easy to extend for count data. However, one could proceed in a two-stage process by first learning the chordal graphs from binarised text through the same scoring functions in Section 9.1.1, and then perform the parameter estimation of Poisson likelihoods from counts, as we did in Section 9.1.2 for the categorical entries. In particular, we have already been developing two estimation strategies that we explain next:

- **Poisson Markov Network.** This strategy consists in learning the maximum likelihood parameters of a multivariate Poisson distribution via an EM algorithm (Karlis, 2003) for each clique and separator in the chordal graph. Then, we use Eq. (9.15) to calculate the joint probability distribution.
- **Poisson Bayesian Network.** This strategy consists in learning a Poisson regression model for each word in the graph with its parents as independent variables. Then, we use Eq. (9.17) to calculate the joint probability distribution.

Apart from PFA, the resulting Poisson chordal models should be compared with other recently proposed graphical models (Inouye et al., 2017) and sum-product networks (Molina et al., 2017) for count data.

Similar to hierarchical latent trees, we also envision the incorporation of global latent variables that interact with a few observed variables and summarise the topical structure of a subset of words. For example, we could proceed by identifying “structural signatures” in the graph which suggest the presence of a hidden variable (Elidan et al., 2001).

## 10.2 Conclusion

Probabilistic topic models have enabled the representation of high-dimensional text into semantically meaningful structures known as topics. Unfortunately, with the advent of new forms of communication, written communication has experienced the shortening of text messages. Topic models, which were not originally thought for short documents, have had troubles to learn meaningful topics on them. Fortunately, these new forms of communication have also brought lots of metadata associated with the message which has become essential for contextualizing the medium. By leveraging on this metadata, a new wave of topic models for short text was developed to learn context-specific topics which became useful for a wide range of tasks. However, most of these topic models still fell into the class of LVMs, which is composed of probability models that use several document-specific latent variables to explain the observed words. The estimation of these local latent variables from little evidence, such as short documents, is challenging and hence, it becomes natural to question their need.

In this thesis, we went from tasks to fundamentals on the exploration of the topical structure of short text. We started by tackling the problem of event detection in Twitter through context-specific topic models. We first showed that we could detect local events by aggregating tweets by hash tag and using these pooled documents in a standard topic model whose topics were used by an extension of DBSCAN (Density-based Spatial Clustering of Applications with Noise) to uncover the events. We then proposed a fully probabilistic event detection method which integrated topic modeling and clustering, so that topics could be learned from clustered documents and clusters could rely on context-specific topics. Before proposing new probability models for text, we addressed the intrinsic evaluation of PFA, a sub-class of LVM which builds on the bagged representation of text. We did that by extending a left-to-right sequential sampler that had previously provided state-of-the-art results in LDA and by proposing several samplers that build proposals from the mean-field approximation to the optimal proposal. Finally, we questioned the need for local latent variables, specially for short text, by proposing new CGMs for binarised text and comparing them to BPFA (Bernoulli PFA) models. The unbiased comparison between different classes of graphical models could be done in terms of likelihood thanks to the samplers presented previously.

Therefore, this thesis has provided new insights, that we summarise next, to the research questions posed in the introduction:

**Can probability models that leverage on contextual information be effective for detecting events in mediums like Twitter?** We conducted this study in a data set of

tweets crawled during the festivities of “La Mercè” in Barcelona and we showed that topic models, a well-known class of probability models for text, must be amended to take contextual information into account and, in that way, mitigate the lack of word co-occurrence in short text documents like tweets. We explored two ways to integrate contextual information with topic models: the heuristic Tweet-SCAN and the probabilistic WARBLE. In the heuristic approach, we pooled tweets by hash tag and we also tried by top keyword. Both pooling strategies were useful to detect events from tweets in “La Mercè”, but the latter was slightly more discriminative than the former with events that overlap in time and space. This preliminary result motivated the development of a fully probabilistic approach that grouped tweets into different components based on their likelihood to belong either to an event or to a non-event component. This allowed tweets, that were grouped into an event component, to be more homogeneous than the heuristic grouping performed earlier, and hence, topics were more discriminative than in the previous heuristic solution. Furthermore, the fully probabilistic approach allowed us to define a spatio-temporal background for non-event tweets which tend to vary smoothly across space and along time. We showed that the presence of this background was also helpful for uncovering local events in “La Mercè” where events were concentrated in space and time. Furthermore, we compared Tweet-SCAN and WARBLE to an existing method in the literature and we showed that both solutions were far better than that proposed by [McInerney and Blei \(2014\)](#) which did not consider spatio-temporal backgrounds and did not jointly perform clustering and topic modelling. Finally, we pointed at multiple directions to keep validating the hypothesis that the use of context is crucial for uncovering events from short text. First, we highlighted the need for publicly available event data sets as well as the existence of labelled events performed by the domain experts or final users. Second, we discussed that WARBLE should be extended with non-parametric Bayesian methods to infer the right number of events and topics from the data, as well as the spatio-temporal densities for the background.

### **Can we develop accurate likelihood estimation methods for PFA topic models?**

The recent finding of a closed-form expression for the marginal likelihood of PFA ([Filstroff et al., 2018](#)) enabled us to validate the accuracy of the proposed estimation methods in downsized setups. For real-world scenarios, the intractability of the closed-form expression hampered the validation of their accuracy, but it did not prevent us from studying their convergence and bounding their accuracy. In particular, we developed two estimation methods/approaches whose accuracy was studied in these terms: the left-to-right sequential sampler and the mean-field variational importance sampling. The former was a non-trivial extension of the state-of-the-art sampler for LDA topic models, which needed to take into account many peculiarities of the Gamma-Poisson construction. Despite the good convergences and accuracy properties compared to simpler methods, the computational cost associated with this method was quadratic in the number of non-zero words. Furthermore, the extension of this sampler for the binarised version of PFA, known as BPFA, did not perform as well as in the original PFA in long text documents due to the much bigger sampling space. These results motivated the development of a family of samplers based on importance sampling with variational mean-field proposals. These samplers did not only have lower computational cost, i.e. linear in the number of non-zero words, but the method with upper-bounded proposal built with the forward KL divergence was superior in accuracy and convergence to the previous methods in both PFA and BPFA. Moreover,

the development of variational importance sampling allowed us to derive a stochastic upper and deterministic lower bound to sandwich the marginal likelihood. These bounds were essential to reason about the accuracy of the proposed methods in realistic setups. Finally, we highlighted the need to derive bounds tighter to the marginal likelihood, given that in some situations the variational bounds were too loose and indecisive. We also suggested that the use of more complex variational distribution and other divergences beyond KL could be interesting ways to extend this family of samplers.

### **Can probability models without local latent variables generalize better for text?**

Thanks to the likelihood estimation methods developed in Part II, we were equipped with a set of tools to assess the generalization of probability models with local latent variables like PFA and BPFA. At that point, we proposed to learn CGMs, an expressive family of probability models without latent variables, on binarised text. The *Chordalysis* algorithm (Petitjean and Webb, 2015b), initially proposed for finding associations in categorical data through chordal graphs, was extended with more appropriate metrics and parameter estimation for prediction tasks on text. CGMs constituted the first representative of probability models without local latent variables. The second was a recently proposed topic model (Chen et al., 2017) which learned tree-structured graphs from binarised text and which incorporated several global latent variables. With these two models, we compared their generalization capabilities against BPFA, through their marginal likelihood as well as in extrinsic tasks such as omni-directional prediction and anomaly detection. The outcome of these experiments pointed at the fact that probability models without local latent variables generalized better to unseen documents in short text, but latent variables models like BPFA were superior in regular text. Furthermore, we saw that the proposed CGMs were superior to hierarchical latent trees in short text and they had similar performance in regular text. Last but not least, *Chordalysis* was able to scale to much bigger data sets than its competitors without compromising the prediction capabilities. However, future work should address the scaling of *Chordalysis* with the vocabulary size, as well as the extension to count data. An approach to address the latter problem was already developed in the future work which consists in learning multivariate or conditional Poisson distributions for each clique or variable in the chordal graph.

In conclusion, this thesis has explored the topical structure of short text from different angles. In the first part, we have studied a concrete application of context-specific topic models for short text, proposing two different techniques to address the problem. In this part, we did not only learn the basics of probability models, but also identified one of the potential causes of the poor generalization of topic models in short text: the local latent variables. In the second part, we equipped ourselves with a toolbox of samplers to evaluate the generalization capabilities of probability models like Poisson factorisation which contain several local latent variables. In the third part, we compared the generalization capabilities of probability models with and without local latent variables and showed that the newly proposed chordal models are specially superior in short text.

The ubiquity of text production and consumption is here to stay as is and the shortening of the message, a consequence of this fast-paced process. With that, the context of communication becomes even more important than the message itself. Therefore, models that are able to accommodate contextual information when available as well as models that intrinsically work well with the lack of context are necessary to answer complex questions

such as “What’s happening?” or “What’s this document about?”. As we have shown, probability models provide a principled and structured way to encode the context into a graph of relationships and groupings, but they are also useful to learn important relationships among words when context is not available. Therefore, this thesis has advanced the use of probability models for short text in situations with and without context, and thus, it has paved the way for current and future forms of written communication.





# Bibliography

- Abdelhaq, H., Sengstock, C., and Gertz, M. (2013). Eventtweet: Online localized event detection from twitter. *Proc. VLDB Endow.*, 6(12):1326–1329.
- Allan, J., Carbonell, J. G., Doddington, G., Yamron, J., and Yang, Y. (1998). Topic detection and tracking pilot study final report.
- Amigó, E., Gonzalo, J., Artiles, J., and Verdejo, F. (2009). A comparison of extrinsic clustering evaluation metrics based on formal constraints. *Information retrieval*, 12(4):461–486.
- Atefeh, F. and Khreich, W. (2015). A survey of techniques for event detection in Twitter. *Computational Intelligence*, 31(1):132–164.
- Bagga, A. and Baldwin, B. (1998). Entity-based cross-document coreferencing using the vector space model. In *Proceedings of the 36th Annual Meeting of the Association for Computational Linguistics and 17th International Conference on Computational Linguistics-Volume 1*, pages 79–85. Association for Computational Linguistics.
- Banfield, J. D. and Raftery, A. E. (1993). Model-based Gaussian and non-Gaussian clustering. *Biometrics*, 49(3):803–821.
- Becker, H., Naaman, M., and Gravano, L. (2011). Beyond trending topics: Real-world event identification on twitter. In *Proceedings of the Fifth International Conference on Weblogs and Social Media, Barcelona, Catalonia, Spain, July 17-21, 2011*.
- Bishop, C. M. (2013). Model-based machine learning. *Phil. Trans. R. Soc. A*, 371(1984):20120222.
- Blei, D. M. (2012). Probabilistic topic models. *Commun. ACM*, 55(4):77–84.
- Blei, D. M. (2014). Build, compute, critique, repeat: Data analysis with latent variable models. *Annual Review of Statistics and Its Application*, 1:203–232.
- Blei, D. M. and Lafferty, J. D. (2006). Dynamic topic models. In *Proceedings of the 23rd international conference on Machine learning*, pages 113–120. ACM.
- Blei, D. M. and Lafferty, J. D. (2007). A correlated topic model of science. *The Annals of Applied Statistics*, 1(1):pp. 17–35.
- Blei, D. M., Ng, A. Y., and Jordan, M. I. (2002). Latent dirichlet allocation. In *Advances in neural information processing systems*, pages 601–608.
- Blei, D. M., Ng, A. Y., and Jordan, M. I. (2003). Latent dirichlet allocation. *Journal of machine Learning research*, 3(Jan):993–1022.

- Boettcher, A. and Lee, D. (2012). Eventradar: A real-time local event detection scheme using Twitter stream. In *Proceedings of the IEEE International Conference on Green Computing and Communications (GreenCom)*, pages 358–367. IEEE.
- Borge-Holthoefer, J., Rivero, A., García, I., Cauhé, E., Ferrer, A., Ferrer, D., Francos, D., Iñiguez, D., Pérez, M. P., Ruiz, G., et al. (2011). Structural and dynamical patterns on online social networks: the spanish may 15th movement as a case study. *PloS one*, 6(8).
- Box, G. E. (1976). Science and statistics. *Journal of the American Statistical Association*, 71(356):791–799.
- Buntine, W. (1991). Theory refinement on bayesian networks. In *Proceedings of the Seventh conference on Uncertainty in Artificial Intelligence*, pages 52–60. Morgan Kaufmann Publishers Inc.
- Buntine, W. (2002). Variational extensions to em and multinomial pca. In *European Conference on Machine Learning*, pages 23–34. Springer.
- Buntine, W. (2009). Estimating likelihoods for topic models. In *Advances in Machine Learning*, pages 51–64, Berlin, Heidelberg. Springer Berlin Heidelberg.
- Buntine, W. and Jakulin, A. (2004). Applying discrete pca in data analysis. In *Proceedings of the 20th conference on Uncertainty in artificial intelligence*, pages 59–66. AUAI Press.
- Buntine, W. and Jakulin, A. (2006). Discrete component analysis. In *Subspace, Latent Structure and Feature Selection*, pages 1–33. Springer.
- Canny, J. (2004). Gap: A factor model for discrete data. In *Proceedings of the 27th Annual International ACM SIGIR Conference on Research and Development in Information Retrieval*, SIGIR '04, pages 122–129, New York, NY, USA. ACM.
- Capdevila, J. (2018). Replication data for: L2r algorithm for likelihood estimation in gap-fa.
- Capdevila, J., Arias, M., and Arratia, A. (2016a). GeoSRS: A hybrid social recommender system for geolocated data. *Information Systems*, 57:111 – 128.
- Capdevila, J., Cerquides, J., Nin, J., and Torres, J. (2015). Tweet-SCAN: An event discovery technique for geo-located tweets. In *Artificial Intelligence Research and Development: Proceedings of the 18th International Conference of the Catalan Association for Artificial Intelligence*, volume 277, page 110. IOS Press.
- Capdevila, J., Cerquides, J., Nin, J., and Torres, J. (2017a). Tweet-SCAN: An event discovery technique for geo-located tweets. *Pattern Recognition Letters*, 93:58 – 68. Pattern Recognition Techniques in Data Mining.
- Capdevila, J., Cerquides, J., and Torres, J. (2016b). Recognizing warblers: a probabilistic model for event detection in twitter. Presented at the Workshop of Anomaly Detection at the International Conference on Machine Learning (ICML).
- Capdevila, J., Cerquides, J., and Torres, J. (2017b). Event detection in location-based social networks. In *Data Science and Big Data: An Environment of Computational Intelligence*, pages 161–186. Springer.

- Capdevila, J., Cerquides, J., and Torres, J. (2018a). Mining urban events from the tweet stream through a probabilistic mixture model. *Data Mining and Knowledge Discovery*, 32(3):764–786.
- Capdevila, J., Cerquides, J., and Torres, J. (2018b). Model-based machine learning for retrospective event detection. Presented at 5th BSC Severo Ochoa Doctoral Symposium.
- Capdevila, J., Cerquides, J., Torres, J., Petitjean, F., and Buntine, W. (2018c). A left-to-right algorithm for likelihood estimation in gamma-poisson factor analysis. In *Joint European Conference on Machine Learning and Knowledge Discovery in Databases*, pages 638–654. Springer.
- Capdevila, J., Pericacho, G., Torres, J., and Cerquides, J. (2016c). Scaling DBSCAN-like algorithms for event detection systems in twitter. In *International Conference on Algorithms and Architectures for Parallel Processing*, volume 10048, pages 356–373. Springer.
- Capdevila, J., Zhao, H., Petitjean, F., and Buntine, W. (2018d). Experiments with learning graphical models on text. *Behaviormetrika*.
- Chandola, V., Banerjee, A., and Kumar, V. (2009). Anomaly detection: A survey. *ACM Comput. Surv.*, 41(3):15:1–15:58.
- Chen, L. and Roy, A. (2009). Event detection from flickr data through wavelet-based spatial analysis. In *Proceedings of the 18th ACM conference on Information and knowledge management*, pages 523–532. ACM.
- Chen, P., Zhang, N. L., Liu, T., Poon, L. K., Chen, Z., and Khawar, F. (2017). Latent tree models for hierarchical topic detection. *Artificial Intelligence*, 250:105 – 124.
- Cheng, T. and Wicks, T. (2014). Event detection using Twitter: a spatio-temporal approach. *PloS one*, 9(6):1–10.
- Choi, M. J., Tan, V. Y., Anandkumar, A., and Willsky, A. S. (2011). Learning latent tree graphical models. *Journal of Machine Learning Research*, 12(May):1771–1812.
- D’Andrea, E., Ducange, P., Lazzerini, B., and Marcelloni, F. (2015). Real-time detection of traffic from twitter stream analysis. *IEEE Transactions on Intelligent Transportation Systems*, 16(4):2269–2283.
- Davis, J. and Goadrich, M. (2006). The relationship between precision-recall and ROC curves. In *Proc. of the 23rd international conference on Machine learning*, pages 233–240. ACM.
- Dieng, A. B., Tran, D., Ranganath, R., Paisley, J., and Blei, D. (2017). Variational inference via  $\chi$  upper bound minimization. In *Advances in Neural Information Processing Systems*, pages 2732–2741.
- Elidan, G., Lotner, N., Friedman, N., and Koller, D. (2001). Discovering hidden variables: A structure-based approach. In *Advances in Neural Information Processing Systems*, pages 479–485.

- Endres, D. M. and Schindelin, J. E. (2003). A new metric for probability distributions. *IEEE Transactions on Information theory*.
- Ester, M., Kriegel, H.-P., Sander, J., and Xu, X. (1996). A density-based algorithm for discovering clusters in large spatial databases with noise. In *Kdd*, volume 96, pages 226–231.
- Ferguson, T. S. (1973). A bayesian analysis of some nonparametric problems. *The annals of statistics*, pages 209–230.
- Filstroff, L., Lumberras, A., and Févotte, C. (2018). Closed-form marginal likelihood in gamma-poisson matrix factorization. In Dy, J. and Krause, A., editors, *Proceedings of the 35th International Conference on Machine Learning*, volume 80 of *Proceedings of Machine Learning Research*, pages 1505–1513. PMLR.
- Friedman, N., Geiger, D., and Goldszmidt, M. (1997). Bayesian network classifiers. *Machine Learning*, 29(2):131–163.
- Garcia-Gasulla, D., Alvarez-Napagao, S., Tejeda-Gómez, A., Oliva-Felipe, L., Vázquez-Salceda, J., Gómez-Sebastià, I., and Bejar, J. (2014). Social network data analysis for event detection. In *21st European Conference on Artificial Intelligence (ECAI2014)*, volume 263, pages 1009–1010. IOS Press.
- Gelman, A., Meng, X.-L., and Stern, H. (1996). Posterior predictive assessment of model fitness via realized discrepancies. *Statistica sinica*, pages 733–760.
- Geman, S. and Geman, D. (1984). Stochastic relaxation, gibbs distributions, and the bayesian restoration of images. *IEEE transactions on pattern analysis and machine intelligence*, 6(6):721.
- Ghahramani, Z. (2015). Probabilistic machine learning and artificial intelligence. *Nature*, 521(7553):452.
- Gopalan, P., Hofman, J. M., and Blei, D. M. (2015). Scalable recommendation with poisson factorization. In *Proceedings of the 31st Conference on Uncertainty in Artificial Intelligence*, pages 326–335.
- Griffiths, T. L. and Steyvers, M. (2004). Finding scientific topics. *Proceedings of the National academy of Sciences*, 101(suppl 1):5228–5235.
- Grosse, R. B., Ghahramani, Z., and Adams, R. P. (2015). Sandwiching the marginal likelihood using bidirectional monte carlo. *arXiv preprint arXiv:1511.02543*.
- He, Q., Chang, K., and Lim, E.-P. (2007). Analyzing feature trajectories for event detection. In *Proceedings of the 30th Annual International ACM SIGIR Conference on Research and Development in Information Retrieval*, SIGIR '07, pages 207–214, New York, NY, USA. ACM.
- Heckerman, D. and Chickering, D. (1995). Learning Bayesian networks: The combination of knowledge and statistical data. In *Machine Learning*, pages 20–197.

- Hinton, G. E. (2002). Training products of experts by minimizing contrastive divergence. *Neural computation*, 14(8):1771–1800.
- Hinton, G. E. and Salakhutdinov, R. R. (2009). Replicated softmax: an undirected topic model. In *Advances in neural information processing systems*, pages 1607–1614.
- Hjort, N. L. et al. (1990). Nonparametric bayes estimators based on beta processes in models for life history data. *The Annals of Statistics*, 18(3):1259–1294.
- Hong, L. and Davison, B. D. (2010). Empirical study of topic modeling in twitter. In *Proceedings of the First Workshop on Social Media Analytics, SOMA '10*, pages 80–88, New York, NY, USA. ACM.
- Hornik, K., Leisch, F., and Zeileis, A. (2003). Jags: A program for analysis of bayesian graphical models using gibbs sampling. In *Proceedings of DSC*, volume 2, pages 1–1.
- Hu, C., Rai, P., and Carin, L. (2016). Non-negative matrix factorization for discrete data with hierarchical side-information. In *Artificial Intelligence and Statistics*, pages 1124–1132.
- Inouye, D. I., Yang, E., Allen, G. I., and Ravikumar, P. (2017). A review of multivariate distributions for count data derived from the poisson distribution. *Wiley Interdisciplinary Reviews: Computational Statistics*, 9(3):e1398.
- Ji, C., Shen, H., and West, M. (2010). Bounded approximations for marginal likelihoods.
- Juan, A. and Vidal, E. (2002). On the use of bernoulli mixture models for text classification. *Pattern Recognition*, 35(12):2705 – 2710. Pattern Recognition in Information Systems.
- Karlis, D. (2003). An em algorithm for multivariate poisson distribution and related models. *Journal of Applied Statistics*, 30(1):63–77.
- Kim, E.-K., Seok, J. H., Oh, J. S., Lee, H. W., and Kim, K. H. (2013). Use of hangeul twitter to track and predict human influenza infection. *PloS one*, 8(7):e69305.
- Kingma, D. P. and Welling, M. (2014). Auto-encoding variational bayes. In *The 2nd International Conference on Learning Representations (ICLR)*.
- Koller, D. and Friedman, N. (2009a). *Probabilistic graphical models: principles and techniques*. MIT press.
- Koller, D. and Friedman, N. (2009b). *Probabilistic Graphical Models: Principles and Techniques - Adaptive Computation and Machine Learning*. The MIT Press.
- Kontkanen, P. and Myllymäki, P. (2007). A linear-time algorithm for computing the multinomial stochastic complexity. *Information Processing Letters*, 103(6):227–233.
- Krumm, J. and Horvitz, E. (2015). Eyewitness: Identifying local events via space-time signals in Twitter feeds. In *Proceedings of the 23rd SIGSPATIAL International Conference on Advances in Geographic Information Systems*. ACM.

- Kulldorff, M. (1997). A spatial scan statistic. *Communications in Statistics-Theory and methods*, 26(6):1481–1496.
- Kulldorff, M., Heffernan, R., Hartman, J., Assunção, R., and Mostashari, F. (2005). A space-time permutation Scan statistic for disease outbreak detection. *PLoS Med*, 2(3).
- Lee, C.-H. (2012). Mining spatio-temporal information on microblogging streams using a density-based online clustering method. *Expert Systems with Applications*, 39(10):9623 – 9641.
- Lee, R. and Sumiya, K. (2010). Measuring geographical regularities of crowd behaviors for Twitter-based geo-social event detection. In *Proceedings of the 2nd ACM SIGSPATIAL International Workshop on Location Based Social Networks (LBSN)*, pages 1–10.
- Li, C., Wang, H., Zhang, Z., Sun, A., and Ma, Z. (2016). Topic modeling for short texts with auxiliary word embeddings. In *Proc. of the 39th International ACM SIGIR conference on Research and Development in Information Retrieval*, pages 165–174.
- Li, L., Goodchild, M. F., and Xu, B. (2013). Spatial, temporal, and socioeconomic patterns in the use of twitter and flickr. *Cartography and Geographic Information Science*, 40(2):61–77.
- Li, R., Lei, K. H., Khadiwala, R., and Chang, K. C. C. (2012). Tedas: A twitter-based event detection and analysis system. In *2012 IEEE 28th International Conference on Data Engineering*, pages 1273–1276.
- Li, Z., Wang, B., Li, M., and Ma, W.-Y. (2005). A probabilistic model for retrospective news event detection. In *Proceedings of the 28th annual international ACM SIGIR conference on Research and development in information retrieval*, pages 106–113. ACM.
- Lim, K. W., Buntine, W., Chen, C., and Du, L. (2016). Nonparametric bayesian topic modelling with the hierarchical pitman-yor processes. *Int. J. Approx. Reasoning*, 78(C):172–191.
- Liu, T., Zhang, N. L., and Chen, P. (2014). Hierarchical latent tree analysis for topic detection. In *Joint European conference on machine learning and knowledge discovery in databases*, pages 256–272. Springer.
- Liu, Z., Malone, B., and Yuan, C. (2012). Empirical evaluation of scoring functions for bayesian network model selection. In *BMC bioinformatics*, volume 13, page S14. BioMed Central.
- Long, R., Wang, H., Chen, Y., Jin, O., and Yu, Y. (2011). Towards effective event detection, tracking and summarization on microblog data. In *Web-Age Information Management*, pages 652–663. Springer.
- Lunn, D., Jackson, C., Best, N., Thomas, A., and Spiegelhalter, D. (2012). *The BUGS book: A practical introduction to Bayesian analysis*. CRC press.
- Maurya, A., Murray, K., Liu, Y., Dyer, C., Cohen, W. W., and Neill, D. B. (2016). Semantic scan: Detecting subtle, spatially localized events in text streams. *CoRR*, abs/1602.04393.

- McCreadie, R., Soboroff, I., Lin, J., Macdonald, C., Ounis, I., and McCullough, D. (2012). On building a reusable twitter corpus. In *Proceedings of the 35th International ACM SIGIR Conference on Research and Development in Information Retrieval*, SIGIR '12, pages 1113–1114, New York, NY, USA. ACM.
- McInerney, J. and Blei, D. M. (2014). Discovering newsworthy tweets with a geographical topic model. *NewsKDD: Data Science for News Publishing workshop. Workshop in conjunction with KDD2014 the 20th ACM SIGKDD Conference on Knowledge Discovery and Data Mining*.
- McMinn, A. J., Moshfeghi, Y., and Jose, J. M. (2013). Building a large-scale corpus for evaluating event detection on twitter. In *Proceedings of the 22nd ACM international conference on Information & Knowledge Management*, pages 409–418. ACM.
- Mimno, D. and Blei, D. (2011). Bayesian checking for topic models. In *Proceedings of the Conference on Empirical Methods in Natural Language Processing*, pages 227–237. Association for Computational Linguistics.
- Minka, T., Winn, J., Guiver, J., and Knowles, D. (2012). Infer .net 2.5, 2012. *Microsoft Research Cambridge*.
- Mitchell, T. M. (1997). *Machine Learning*. McGraw-Hill, New York.
- Molina, A., Natarajan, S., and Kersting, K. (2017). Poisson sum-product networks: A deep architecture for tractable multivariate poisson distributions. In *AAAI*, pages 2357–2363.
- Murphy, K. P. (2012). *Machine Learning: A Probabilistic Perspective*. The MIT Press.
- Murray, I. and Salakhutdinov, R. R. (2009). Evaluating probabilities under high-dimensional latent variable models. In *Advances in neural information processing systems*, pages 1137–1144.
- Neal, R. M. (2001). Annealed importance sampling. *Statistics and computing*, 11(2):125–139.
- Newman, N. (2011). Mainstream media and the distribution of news in the age of social discovery. *Reuters Institute for the Study of Journalism, University of Oxford*.
- Newton, M. A. and Raftery, A. E. (1994). Approximate bayesian inference with the weighted likelihood bootstrap. *Journal of the Royal Statistical Society. Series B (Methodological)*, pages 3–48.
- Nguyen, D., Billingsley, R., Du, L., and Johnson, M. (2015). Improving topic models with latent feature word representations. *Trans. of the Association for Computational Linguistics*, 3:299–313.
- Nigam, K., McCallum, A. K., Thrun, S., and Mitchell, T. (2000). Text classification from labeled and unlabeled documents using em. *Machine learning*, 39(2-3):103–134.
- Ogihara, Z. P., Zaki, M., Parthasarathy, S., Ogihara, M., and Li, W. (1997). New algorithms for fast discovery of association rules. In *In 3rd Intl. Conf. on Knowledge Discovery and Data Mining*. Citeseer.

- Oliveira, V. D. (2013). Hierarchical poisson models for spatial count data. *Journal of Multivariate Analysis*, 122:393 – 408.
- Paisley, J., Wang, C., Blei, D., and Jordan, M. (2015). Nested hierarchical Dirichlet processes. *IEEE Trans. on Pattern Analysis and Machine Intelligence*, 37(2):256–270.
- Panagiotou, N., Katakis, I., and Gunopulos, D. (2016). Detecting events in online social networks: Definitions, trends and challenges. In *Solving Large Scale Learning Tasks. Challenges and Algorithms*, pages 42–84. Springer.
- Petitjean, F., Allison, L., Webb, G. I., and Nicholson, A. E. (2014). A statistically efficient and scalable method for log-linear analysis of high-dimensional data. In *IEEE International Conference on Data Mining*, pages 480–489.
- Petitjean, F. and Webb, G. I. (2015a). Scaling log-linear analysis to datasets with thousands of variables. In *SIAM International Conference on Data Mining*, pages 469–477.
- Petitjean, F. and Webb, G. I. (2015b). Scaling log-linear analysis to datasets with thousands of variables. In *SIAM International Conference on Data Mining*, pages 469–477.
- Petitjean, F., Webb, G. I., and Nicholson, A. E. (2013). Scaling log-linear analysis to high-dimensional data. In *IEEE International Conference on Data Mining*, pages 597–606.
- Quan, X., Kit, C., Ge, Y., and Pan, S. J. (2015). Short and sparse text topic modeling via self-aggregation. In *Proceedings of the 24th International Conference on Artificial Intelligence (IJCAI)*.
- Sakaki, T., Okazaki, M., and Matsuo, Y. (2010). Earthquake shakes twitter users: real-time event detection by social sensors. In *Proceedings of the 19th international conference on World wide web*, pages 851–860. ACM.
- Salton, G. and Buckley, C. (1988). Term-weighting approaches in automatic text retrieval. *Inf. Process. Manage.*, 24(5):513–523.
- Salton, G., Wong, A., and Yang, C.-S. (1975). A vector space model for automatic indexing. *Communications of the ACM*, 18(11):613–620.
- Sander, J., Ester, M., Kriegel, H.-P., and Xu, X. (1998). Density-based clustering in spatial databases: The algorithm gdbscan and its applications. *Data Mining and Knowledge Discovery*, 2(2):169–194.
- Saul, L. K. and Jordan, M. I. (1996). Exploiting tractable substructures in intractable networks. In *Advances in neural information processing systems*, pages 486–492.
- Schwarz, G. et al. (1978). Estimating the dimension of a model. *The annals of statistics*, 6(2):461–464.
- Sethuraman, J. (1994). A constructive definition of dirichlet priors. *Statistica sinica*, pages 639–650.
- Sibuya, M., Yoshimura, I., and Shimizu, R. (1964). Negative multinomial distribution. *Annals of the Institute of Statistical Mathematics*, 16(1):409–426.



- Silander, T., Leppä-aho, J., Jääsaari, E., and Roos, T. (2018). Quotient normalized maximum likelihood criterion for learning bayesian network structures. In Storkey, A. and Perez-Cruz, F., editors, *Proceedings of the Twenty-First International Conference on Artificial Intelligence and Statistics*, volume 84 of *Proceedings of Machine Learning Research*, pages 948–957, Playa Blanca, Lanzarote, Canary Islands. PMLR.
- Singh, S. (2015). Spatial temporal analysis of social media data. Master’s thesis, Technische Universität München.
- Steinbach, M., Karypis, G., Kumar, V., et al. (2000). A comparison of document clustering techniques. In *KDD workshop on text mining*, volume 400, pages 525–526. Boston.
- Stelter, B. and Cohen, N. (2008). Citizen journalists provided glimpses of mumbai attacks.
- Suzuki, J. and Kawahara, J. (2017). Branch and bound for regular Bayesian network structure learning. In *Conference on Uncertainty in Artificial Intelligence*, Sydney, Australia.
- Teh, Y., Jordan, M., Beal, M., and Blei, D. (2006a). Hierarchical Dirichlet processes. *Journal of the ASA*, 101(476):1566–1581.
- Teh, Y. W., Jordan, M. I., Beal, M. J., and Blei, D. M. (2006b). Hierarchical dirichlet processes. *Journal of the american statistical association*, 101(476).
- Thibaux, R. and Jordan, M. I. (2007). Hierarchical beta processes and the indian buffet process. In *Artificial Intelligence and Statistics*, pages 564–571.
- Tran, D., Kucukelbir, A., Dieng, A. B., Rudolph, M., Liang, D., and Blei, D. M. (2016). Edward: A library for probabilistic modeling, inference, and criticism. *arXiv preprint arXiv:1610.09787*.
- Van Rijsbergen, C. J. (1974). Foundation of evaluation. *Journal of Documentation*, 30(4):365–373.
- Wainwright, M. J., Jordan, M. I., et al. (2008). Graphical models, exponential families, and variational inference. *Foundations and Trends® in Machine Learning*, 1(1–2):1–305.
- Wallach, H. M. (2006). Topic modeling: beyond bag-of-words. In *Proceedings of the 23rd international conference on Machine learning*, pages 977–984. ACM.
- Wallach, H. M., Mimno, D. M., and McCallum, A. (2009a). Rethinking lda: Why priors matter. In *Advances in neural information processing systems*, pages 1973–1981.
- Wallach, H. M., Murray, I., Salakhutdinov, R., and Mimno, D. (2009b). Evaluation methods for topic models. In *Proceedings of the 26th Annual International Conference on Machine Learning*, ICML ’09, pages 1105–1112, New York, NY, USA. ACM.
- Wallach, H. M., Murray, I., Salakhutdinov, R., and Mimno, D. (2009c). Evaluation methods for topic models. In *Proceedings of the 26th annual international conference on machine learning*, pages 1105–1112. ACM.
- Walther, M. and Kaisser, M. (2013). Geo-spatial event detection in the twitter stream. In *European conference on information retrieval*, pages 356–367. Springer.

- Watanabe, K., Ochi, M., Okabe, M., and Onai, R. (2011). Jasmine: a real-time local-event detection system based on geolocation information propagated to microblogs. In *Proceedings of the 20th ACM international conference on Information and knowledge management*, pages 2541–2544. ACM.
- Webb, G. I. (2008). Layered critical values: a powerful direct-adjustment approach to discovering significant patterns. *Machine Learning*, 71(2-3):307–323.
- Webb, G. I. and Petitjean, F. (2016). A multiple test correction for streams and cascades of statistical hypothesis tests. In *ACM SIGKDD International Conference on Knowledge Discovery and Data Mining*, pages 1225–1264.
- Weng, J. and Lee, B.-S. (2011). Event detection in twitter. *ICWSM*, 11:401–408.
- Wong, W.-K., Moore, A., Cooper, G., and Wagner, M. (2005). What’s strange about recent events (wsare): An algorithm for the early detection of disease outbreaks. *Journal of Machine Learning Research*, 6(Dec):1961–1998.
- Wong, W.-K. and Neill, D. B. (2009). Tutorial on event detection kdd.
- Xie, P. and Xing, E. P. (2013). Integrating document clustering and topic modeling. In *Proceedings of the Twenty-Ninth Conference on Uncertainty in Artificial Intelligence*, pages 694–703. AUAI Press.
- Yin, J. and Wang, J. (2014). A Dirichlet multinomial mixture model-based approach for short text clustering. In *Proceedings of the 20th ACM SIGKDD international conference on Knowledge discovery and data mining*, pages 233–242. ACM.
- Zhang, C., Zhou, G., Yuan, Q., Zhuang, H., Zheng, Y., Kaplan, L., Wang, S., and Han, J. (2016a). Geoburst: Real-time local event detection in geo-tagged tweet streams. In *Proceedings of the 39th International ACM SIGIR conference on Research and Development in Information Retrieval*, pages 513–522. ACM.
- Zhang, L., Sun, X., and Zhuge, H. (2013). Location-driven geographical topic discovery. In *Semantics, Knowledge and Grids (SKG), 2013 Ninth International Conference on*, pages 210–213. IEEE.
- Zhang, Y., Zhao, Y., David, L., Henao, R., and Carin, L. (2016b). Dynamic poisson factor analysis. In *Data Mining (ICDM), 2016 IEEE 16th International Conference on*, pages 1359–1364. IEEE.
- Zhou, M. (2015). Infinite edge partition models for overlapping community detection and link prediction. In Lebanon, G. and Vishwanathan, S. V. N., editors, *Proceedings of the Eighteenth International Conference on Artificial Intelligence and Statistics*, volume 38 of *Proceedings of Machine Learning Research*, pages 1135–1143, San Diego, California, USA. PMLR.
- Zhou, M. and Carin, L. (2015). Negative binomial process count and mixture modeling. *IEEE Transactions on Pattern Analysis and Machine Intelligence*, 37(2):307–320.

- Zhou, M., Hannah, L., Dunson, D., and Carin, L. (2012). Beta-negative binomial process and poisson factor analysis. In Lawrence, N. D. and Girolami, M., editors, *Proceedings of the Fifteenth International Conference on Artificial Intelligence and Statistics*, volume 22 of *Proceedings of Machine Learning Research*, pages 1462–1471, La Palma, Canary Islands. PMLR.



# Acronyms

**AIS** Annealed Importance Sampling

**API** Application Programming Interface

**AUC** Area Under Curve

**AUC-PR** Area Under Curve - Precision Recall

**AUC-ROC** Area Under Curve - Receiver Operating Curve

**BDeu** Bayesian Dirichlet equivalent uniform

**BerPo** Bernoulli-Poisson

**BIC** Bayesian Information Criterion

**BN** Bayesian Network

**BNB** Beta-Negative Binomial Process

**BP** Beta Process

**BPFA** Bernoulli PFA

**CGM** Chordal Graphical Model

**CPT** Conditional Probability Table

**DAG** Directed Acyclic Graph

**DBSCAN** Density-based Spatial Clustering of Applications with Noise

**DGM** Directed Graphical Model

**DP** Dirichlet Process

**DS** Direct Sampling

**ELBO** Evidence Lower Bound

**EUBO** Evidence Upper Bound

**GaP** Gamma Poisson

**GDBSCAN** Generalised Density-based Spatial Clustering of Applications with Noise

**GIS** Geographical Information System

**GMM** Gaussian Mixture Model

**HDP** Hierarchical Dirichlet Process

**HLTA** Hierarchical latent tree analysis

**HLTM** Hierarchical Latent Tree Model

**HM** Harmonic Mean

**IDF** Inverse Document Frequency

**IS** Importance Sampling

**IS-IP** Iterated Pseudo-Counts

**JPT** Joint Probability Table

**JS** Jensen-Shannon

**KL** Kullback-Leibler

**LB-VIS** Lower-Bounded Variational Importance Sampling

**LBFS** Lexicographic Breadth First Search

**LDA** Latent Dirichlet Allocation

**LLA** Log-Linear Analysis

**LTM** Latent Tree Model

**LVM** Latent Variable Model

**MAP** Maximum a Posteriori

**MCMC** Markov Chain Monte Carlo

**MFI** Mean Field Importance

**MH** Metropolis Hastings

**MN** Markov Network

**MoB** Mixture of Bernoullis

**MoU** Mixture of Unigrams

**mPCA** Multinomial Principal Component Analysis

**NB** Negative Binomial

**NBP** Negative Binomial Process

**NED** New Event Detection

**nHDP** nested Hierarchical Dirichlet Process

**NIW** Normal-Inverse-Wishart

**NM** Negative Multinomial

**NML** Normalized Maximum Likelihood

**PCA** Principal Component Analysis

**PEO** Perfect Elimination Ordering

**PFA** Poisson Factor Analysis

**PGM** Probabilistic Graphical Model

**PPC** Posterior Predictive Checks

**PTM** Probabilistic Topic Models

**qNML** quotient Normalized Maximum Likelihood

**RBM** Restricted Boltzmann Machines

**RED** Retrospective Event Detection

**RMSE** Root Mean Squared Error

**SMT** Subfamilywise Multiple Testing

**STSS** Spatial Scan Statistic

**TDT** Topic Detection and Tracking

**TF-IDF** Term Frequency Inverse Document Frequency

**TREC** Text REtrieval Conference

**UB-VIS** Upper-Bounded Variational Importance Sampling

**UGM** Undirected Graphical Models

**VIS** Variational Importance Sampling





# Appendices



# A

## Notation and terminology

In this appendix, we introduce the mathematical notation and terminology that is used throughout the thesis. In general, we use regular lower case to denote fixed and random scalars and bold lower case to denote vectors. Bold upper case letters denote matrices and non-bold upper case, constants. Greek letters are used to denote random and fixed parameters in statistical distributions and sample estimates.

**General math notation:** to introduce to scalar, vector, matrices and other data structures.

|                       |   |
|-----------------------|---|
| $x$                   | scalar  |
| $\mathbf{x}$          | $[x_1, \dots, x_K]$ vector of size $K$                            |
| $\mathbf{X}$          | $[\mathbf{x}_1, \dots, \mathbf{x}_K]$ matrix of size $N \times K$ |
| $\mathbf{x}_{i:}$     | $i$ -th row vector in $\mathbf{X}$                                |
| $\mathbf{x}_{:j}$     | $j$ -th column vector in $\mathbf{X}$                             |
| $x_{ij}$              | $i, j$ -th scalar entry in $\mathbf{X}$                           |
| $[...]$               | vector and matrix   |
| $(...)$               | sequence  |
| $\langle ... \rangle$ | tuple   |
| $\{...\}$             | set   |






**Probability and statistics notation:** to discuss a wide variety of probability concepts.

|                               |  |
|-------------------------------|--|
| r.v.                          | random variable  |
| $x \perp\!\!\!\perp y$        | r.v. $x$ is independent of r.v. $y$                                |
| $x \perp\!\!\!\perp y \mid z$ | r.v. $x$ is independent of r.v. $y$ given r.v. $z$                 |
| pmf                           | probability mass function  |
| pdf                           | probability density function                                       |
| $p(x; \theta)$                | pdf or pmf of $x$ parametrised according to $\theta$               |
| $\tilde{p}(x; \theta)$        | unnormalized pdf or pmf of $x$ parametrised according to $\theta$  |
| $p(x y; \theta)$              | conditional pdf or pmf of $x$ parametrised according to $\theta$   |
| $p(x, y; \theta)$             | joint pdf or pmf of $x$ and $y$ parametrised according to $\theta$ |
| $x \sim p$                    | $x$ is distributed according to $p$                                |
| $x y \sim p$                  | $x$ conditioned to $y$ is distributed according to $p$             |
| $x^*$                         | held-out observation   |
| $\hat{\theta}$                | point estimate of $\theta$   |

**Graph notation:** is used to introduce probability models in general.

|                  |  |
|------------------|--|
| $\mathcal{G}$    | graph $\mathcal{G} = (\mathcal{V}, \mathcal{E})$               |
| $\mathcal{V}$    | set of nodes $\mathcal{V} = \{1...V\}$                         |
| $V$              | number of nodes $V =  \mathcal{V} $                            |
| $\mathcal{E}$    | set of edges $\mathcal{E} = \{(s, t) : s, t \in \mathcal{V}\}$ |
| $E$              | number of edges $E =  \mathcal{E} $                            |
| DAG              | Directed Acyclic Graph   |
| $\text{par}(s)$  | set of parents of $v \in \mathcal{V}$ in a DAG                 |
| $\text{pred}(s)$ | set of predecessors of $v \in \mathcal{V}$ in a DAG            |
| $\mathcal{C}$    | cliques of a graph   |
| $\mathcal{S}$    | separators of a graph  |
| $\psi_c(\cdot)$  | potential function for clique $c$                              |

**Plate notation:** is used to draw probabilistic graphical models.

|   |   |
|---|---|
|    | observed variable   |
|    | latent variable   |
|   | latent parameter (represented through a letter in the Greek alphabet) |
|  | plate of size $N$   |
|  | statistical dependency  |
| $N$   | model hyperparameter (inside a plate)                                 |
| $\gamma$  | distribution hyperparameter (connected to the related variable)       |

**Common text notation:** is used for probability models for text.

|               |  |
|---------------|--|
| $V$           | vocabulary size                                    |
| $v$           | word   |
| $\mathcal{V}$ | vocabulary set $\mathcal{V} = \{v_1, \dots, v_V\}$ |
| $w$           | index in the in the vocabulary set $\mathcal{V}$   |
| $N$           | number of documents                                |
| $L_n$         | length of the $n$ -th document                     |

**Sequenced-specific notation:** is used for probability models that consider a sequenced representation of the bag of words.

|                |  |
|----------------|--|
| $\mathbf{w}_n$ | $n$ -th document represented as a sequence $(w_{n1}, \dots, w_{nL_n})$                           |
| $m$            | $m$ -th position in a document   |
| $w_{nm}$       | $n, m$ -th index in the vocabulary set $\mathcal{V}$   |
| $\mathbf{W}$   | document collection represented as a sequence of sequences $(\mathbf{w}_1, \dots, \mathbf{w}_N)$ |

**Bagged-specific notation:** is used for probability models that consider a bagged representation of the bag of words.

|                   |   |
|-------------------|---|
| $\mathbf{y}_n$ :  | $n$ -th document represented as a vector $[y_{n1}, \dots, y_{nV}] \in \mathbb{N}_0^V$                                 |
| $y_{np}$          | counts of the word $p$ -th word in $\mathcal{V}$ in the $n$ -th document  |
| $\mathbf{Y}$      | document collection represented as a matrix $[\mathbf{y}_{1:}, \dots, \mathbf{y}_{N:}] \in \mathbb{N}_0^{N \times V}$ |
| $\mathbf{y}_{:p}$ | $p$ -th word vector   |
| $y_{np}$          | $n, p$ -th scalar entry   |
| $y_n$ .           | document length ( $L_n$ ), i.e. sum over words  |
| $y_{.p}$          | total word counts in the collection, i.e. sum over documents  |

# B

## Probability Distributions

### B.1 Elemental Probability Distributions

#### B.1.1 Discrete Distributions

**Bernoulli Distribution** is a discrete distribution that expresses the probability of a binary event that is  $x = 1$  with probability  $p$  and  $x = 0$  with probability  $1 - p$ . It describes the outcome of a coin toss with heads' probability  $p$  and tails'  $1 - p$ . The probability mass function (pmf) can be expressed as,

$$\text{Ber}(x; p) = p^x(1 - p)^{1-x} \quad (\text{B.1})$$

where  $0 < p < 1$ ,  $p \in \mathbb{R}$  and the support for this probability distribution is  $x \in \{0, 1\}$ . Besides, the mean of a Bernoulli random variable is  $p$  and its variance  $p(1 - p)$ .

**Binomial Distribution** is a discrete distribution that expresses the probability of  $x$  successes after  $n$  Bernoulli trials with probability of success  $p$ . It describes the outcome of  $n$  coin tosses with heads' probability  $p$  and tails'  $1 - p$ . The pmf can be expressed as,

$$\text{Bin}(x; n, p) = \binom{n}{x} p^x(1 - p)^{n-x} \quad (\text{B.2})$$

where  $0 < p < 1$ ,  $p \in \mathbb{R}$  and the support for this probability distribution is  $x \in \mathbb{N}_0$ . Note that the binomial coefficient  $\binom{n}{x}$  accounts for the number of ways to choose  $x$  items from  $n$  trials. Besides, the mean of a Binomial random variable is  $np$  and its variance  $np(1 - p)$ .

**Categorical Distribution** or Multinoulli distribution is the generalisation of the Bernoulli distribution to  $K$  outcomes  $\mathbf{x} = (x_1, x_2, \dots, x_K)$  with probabilities  $\mathbf{p} = (p_1, p_2, \dots, p_K)$ . It describes the outcome of throwing a  $K$ -sided dice with side probabilities  $\mathbf{p}$ . The pmf can be expressed as,

$$\text{Cat}(\mathbf{x}; \mathbf{p}) = \prod_{k=1}^K p_k^{\mathbb{I}(x_k=1)} \quad (\text{B.3})$$

where  $0 < p_k < 1$ ,  $p_k \in \mathbb{R}$  and the support for this probability distribution is  $x_k \in \{0, 1\}$ . The mean of the  $k$ -th component is  $p_k$  and its variance  $p_k(1 - p_k)$ .

**Multinomial Distribution** is a multivariate discrete distribution that expresses the probability of  $\mathbf{x} = (x_1, x_2, \dots, x_K)$  successes after  $n$  Categorical trials with probabilities  $\mathbf{p} = (p_1, p_2, \dots, p_K)$ . It is also the generalization of the Binomial distribution to  $K$  outcomes with probabilities  $\mathbf{p}$ . It describes the outcomes of  $n$  throws of a  $K$ -sided dice with probabilities  $\mathbf{p}$ . The pmf can be expressed as,

$$\text{Mult}(\mathbf{x}; n, \mathbf{p}) = \binom{n}{x_1 \dots x_K} \prod_{k=1}^K p_k^{x_k} \quad (\text{B.4})$$

where  $0 < p_k < 1$ ,  $p_k \in \mathbb{R}$  and the support for this probability distribution is  $x_k \in \mathbb{N}_0$ . Note that the multinomial coefficient  $\binom{n}{x_1 \dots x_K}$  accounts for the number of ways to divide the whole set of size  $n$  items into subsets with sizes  $x_1 \dots x_K$ . The mean of the  $k$ -th component is  $np_k$  and its variance  $np_k(1 - p_k)$ .

**Poisson Distribution** is a univariate discrete distribution that expresses the number of events  $x$  happening in a fix interval knowing that events occur with constant rate  $\lambda$  and independently of the the latest event. It is often used to describe temporal events like the number of phone calls received by a call centre or radioactive decay. The pmf can be expressed as,

$$\text{Pois}(x; \lambda) = e^{-\lambda} \frac{\lambda^x}{x!} \quad (\text{B.5})$$

where  $\lambda > 0$   $\lambda \in \mathbb{R}$  and the support for this probability distribution is  $x \in \mathbb{N}_0$ . The mean and variance of a Poisson r. v. is  $\lambda$ , the same for both moments.

**Zero-Truncated Poisson Distribution** is a univariate discrete distribution with support in the positive integers. It is a conditional distribution of a Poisson r.v. whose value is positive. The pmf is given by,

$$\text{Pois}_+(x; \lambda) = e^{-\lambda} \frac{\lambda^x}{x!} \quad (\text{B.6})$$

where  $\lambda$  corresponds to the same rate than in the Poisson distribution. A Zero-Truncated Poisson r.v. has mean  $\frac{\lambda e^\lambda}{e^\lambda - 1}$  and variance  $\frac{\lambda + \lambda^2}{1 - e^{-\lambda}} - \frac{\lambda^2}{(1 - e^{-\lambda})^2}$ .

## B.1.2 Continuous Distributions

**Beta Distribution** is a univariate continuous distribution with support over the interval  $[0, 1]$  and parametrised with two positives values  $\alpha$  and  $\beta$ . The probability density function (pdf) can be expressed as,

$$\text{Beta}(x; \alpha, \beta) = \frac{1}{B(\alpha, \beta)} x^{\alpha-1} (1-x)^{\beta-1} \quad (\text{B.7})$$

where  $B(\alpha, \beta) = \frac{\Gamma(\alpha)\Gamma(\beta)}{\Gamma(\alpha+\beta)}$  is the Beta function and  $\Gamma(\cdot)$  the Gamma function. This distribution is commonly used to model the randomness of the proportion  $p$  in the Bernoulli, Binomial and Negative Binomial distributions, amongst others, since it is conjugated to them.



**Dirichlet Distribution** is the multivariate generalisation of the Beta distribution to  $K$  dimensions with support in the  $K - 1$  simplex for  $K \geq 2$  and parametrised with a vector of positives values  $\boldsymbol{\alpha} = \{\alpha_1, \dots, \alpha_K\}$ . The pdf can be expressed as,

$$\text{Dir}(\mathbf{x}; \boldsymbol{\alpha}) = \frac{1}{B(\boldsymbol{\alpha})} \prod_{k=1}^K x_k^{\alpha_k - 1} \quad (\text{B.8})$$

where  $B(\boldsymbol{\alpha}) = \frac{\prod_{k=1}^K \Gamma(\alpha_k)}{\Gamma(\sum_{k=1}^K \alpha_k)}$  is the generalised Beta function to  $K$  dimensions. This distributions is commonly used to model the randomness of proportions in the Categorical and Multinomial distributions, amongst others, since it is conjugated to them.

**Gamma Distribution** is a univariate continuous distribution with support in the interval  $(0, \infty)$  and parametrised with two positives values referred to as scale  $k$  and shape  $\theta$ . The pdf can be expressed as,

$$\text{Ga}(x) = \frac{1}{\Gamma(k)\theta^k} x^{k-1} e^{-\frac{x}{\theta}} \quad (\text{B.9})$$

where  $\Gamma(k)$  refers to the Gamma function. This distribution is commonly used to model the randomness of the rate in Poisson and Exponential distributions, amongst others, since it is conjugated to them.

## B.2 Table of distribution in the exponential family

## B.3 Compound Probability Distributions

Compound or composite distributions are the result of considering that some of the parameters of an elemental probability are random variables distributed according to another elemental distribution. These random variables are then marginalised out and new parametrised distributions emerges.

**Dirichlet-Multinomial Distribution** is a multivariate discrete distribution on the finite support of non-negative integers  $\mathbf{x} = \{x_1, \dots, x_K\}$  which add up to the number of trials  $n$  and parametrised with a positive parameter vector  $\boldsymbol{\alpha} = \{\alpha_1, \dots, \alpha_K\}$ . The p.m.f. has the following parametric form,

$$\text{DirMult}(\mathbf{x}|n, \boldsymbol{\alpha}) = \frac{n! \Gamma(n + \sum_k \alpha_k)}{\Gamma(\sum_k \alpha_k)} \prod_{k=1}^K \frac{\Gamma(x_k + \alpha_k)}{x_k! \Gamma(\alpha_k)} \quad (\text{B.10})$$

The mean of the  $k$ -th dimension is given by  $n \frac{\alpha_k}{\sum_k \alpha_k}$  and its variance, by  $n \frac{\alpha_k}{\sum_i \alpha_i} (1 - \frac{\alpha_k}{\sum_i \alpha_i}) \frac{n + \sum_i \alpha_i}{1 + \sum_i \alpha_i}$ . This compound distribution is built by considering a Multinomial distribution whose probabilities  $\mathbf{p}$  follows a Dirichlet distribution,

$$\text{DirMult}(\mathbf{x}|n, \boldsymbol{\alpha}) = \int \text{Mult}(\mathbf{x}; n, \mathbf{p}) \text{Dir}(\mathbf{p}; \boldsymbol{\alpha}) d\mathbf{p}. \quad (\text{B.11})$$

and then integrating out the probabilities  $\mathbf{p}$ .

| Distribution                  | Parameter(s)<br>$\theta$                       | Natural<br>Parameter(s)<br>$\eta = \eta(\theta)$               | Base<br>Measure<br>$h(x)$ | Sufficient<br>Statistics<br>$t(x)$   | Log-partition<br>$A(\eta)$   |
|-------------------------------|--|--|---------------------------|--|--|
| Bernoulli                     | $p$  | $\log \frac{p}{1-p}$   | 1                         | $x$  | $\log(1 + e^\eta)$   |
| Binomial<br>with known $n$    | $p$  | $\log \frac{p}{1-p}$   | $\binom{n}{x}$            | $x$  | $n \log(1 + e^\eta)$   |
| Poisson                       | $\lambda$                                      | $\log \lambda$   | $\frac{1}{x!}$            | $x$  | $e^\eta$   |
| Categorical                   | $p_1, \dots, p_k$<br>where<br>$\sum_k p_k = 1$ | $\begin{bmatrix} \log p_1 \\ \vdots \\ \log p_k \end{bmatrix}$ | 1                         | $\begin{bmatrix} \mathbb{I}(x=1) \\ \vdots \\ \mathbb{I}(x=k) \end{bmatrix}$ | 0  |
| Multinomial<br>with known $n$ | $p_1, \dots, p_k$<br>where<br>$\sum_k p_k = 1$ | $\begin{bmatrix} \log p_1 \\ \vdots \\ \log p_k \end{bmatrix}$ | $\frac{n!}{\prod_k x_k!}$ | $\begin{bmatrix} x_1 \\ \vdots \\ x_k \end{bmatrix}$                         | 0  |
| Beta                          | $\alpha, \beta$                                | $\begin{bmatrix} \alpha \\ \beta \end{bmatrix}$                | $\frac{1}{x(1-x)}$        | $\begin{bmatrix} \log x \\ \log(1-x) \end{bmatrix}$                          | $\log \Gamma(\eta_1) + \log \Gamma(\eta_2) - \log \Gamma(\eta_1 + \eta_2)$ |
| Dirichlet                     | $\alpha_1, \dots, \alpha_k$                    | $\begin{bmatrix} \alpha_1 \\ \vdots \\ \alpha_k \end{bmatrix}$ | $\frac{1}{\prod_k x_k!}$  | $\begin{bmatrix} \log x_1 \\ \vdots \\ \log x_k \end{bmatrix}$               | $\sum_k \log \Gamma(\eta_k) - \log \Gamma(\sum_k \eta_k)$                  |
| Gamma                         | $k, \theta$                                    | $\begin{bmatrix} k-1 \\ -\frac{1}{\theta} \end{bmatrix}$       | 1                         | $\begin{bmatrix} \log x \\ x \end{bmatrix}$                                  | $\log \Gamma(\eta_1 + 1) - (\eta_1 + 1) \log(-\eta_2)$                     |

Table B.1: Basic probability distributions in Exponential family form.

**Negative Binomial Distribution** is a discrete distribution for the number of successes in a sequence of i.i.d Bernoulli trials with probability  $p$  after observing a given number of  $r$  failures. The p.m.f. can be expressed as,

$$\text{NB}(x|r, p) = \frac{\Gamma(r+x)}{\Gamma(r)} (1-p)^r p^x \quad (\text{B.12})$$

This distribution is used to model over-dispersed counts since its variance  $\frac{rp}{(1-p)^2}$  is higher than its mean  $\frac{rp}{1-p}$ .

As shown in (Zhou et al., 2012), the NB can be constructed by marginalizing a Poisson distribution whose rate  $\theta$  is controlled by a gamma random variable parameterized with shape  $r$  and scale  $\frac{p}{1-p}$  as above. In other words, we can build a NB distribution by,

$$\text{NB}(x; r, p) = \int \text{Pois}(x|\theta) \text{Ga}(\theta; r, \frac{p}{1-p}) d\theta. \quad (\text{B.13})$$

**Negative Multinomial Distribution** (Sibuya et al., 1964) is the multivariate generalization of the NB distribution to  $W$  outcomes ( $W > 1$ ), each occurring with probability  $q_w$  and for a given number of failures  $r$ .

$$\text{NM}(\mathbf{x}|r, \mathbf{q}) = \frac{\Gamma(r + \sum_w x_w)}{\Gamma(r)} (1 - \sum_w p_w)^r \prod_w \frac{q_w^{x_w}}{x_w!} \quad (\text{B.14})$$

As shown in (Filstroff et al., 2018), the NM can be built by marginalizing  $W$  independent Poisson distributions whose rate is controlled by a gamma random variable  $\theta$  that is scaled by a vector  $\phi_\cdot$  of length  $W$ . This can be expressed mathematically as,

$$\text{NM}(x_\cdot; r, q_\cdot = \frac{p\phi_\cdot}{1-p+p\sum_w \phi_w}) = \int \prod_w \text{Pois}(x_w|\theta\phi_w) \text{Ga}(\theta; r, \frac{p}{1-p}) d\theta \quad (\text{B.15})$$

where  $r$  are the number of failures and  $q_\cdot = \frac{p\phi_\cdot}{1-p+p\sum_w \phi_w}$  is the vector of  $W$  success probabilities. When  $\phi_\cdot$  is a probability vector, which sums up to 1, the success probabilities of the NM become  $q_\cdot = p\phi_\cdot$ .

## B.4 Stick-breaking construction

This section describes the stick-breaking process used in the constructive definition of Dirichlet Process provided by (Sethuraman, 1994) to draw the proportions  $(\pi_k)_{k=1}^\infty$ . The process goes as follows. It first draws a collection of Beta random variables,

$$v_k \sim \text{Beta}(1, \alpha) \quad k \in \{1, 2, \dots\} \quad (\text{B.16})$$

where  $\text{Beta}(1, \alpha)$  is parameterized as in Eq. (B.7). Then, this collection of Beta variables is used to create the sequence  $(\pi_k)_{k=1}^\infty$ , a.k.a. stick proportions, through the formula,

$$\pi_k = v_k \prod_{i=1}^{k-1} (1 - v_i) \quad (\text{B.17})$$

which guarantees  $\sum_{k=1}^\infty \pi_k = 1$ . The distribution of stick proportions is known as GEM named after Griffiths, Engen and McCloskey.





# Variational updates for Warble

## C.1 Introduction

This appendix presents the variational inference updates for the mean-field approximation of the WARBLE model depicted in Fig. 5.4. The mean-field approximation assumes the factorized distribution in Eq. (5.14). As introduced earlier, the goal of variational inference is to minimize the KL divergence between the posterior distribution  $p(X|D;\Gamma)$  and the factorized distribution  $q(X;\eta)$ , through a coordinate ascent algorithm that sequentially updates each variable at a time. The variational update for a random variable  $x$  whatsoever can be found by solving Eq. (5.15). Next, we present the functional forms and parameter updates for each variational distribution.

## C.2 Mixture Proportions $q(\pi)$

We can derive the variational distribution for the mixture proportions  $q(\pi)$  by applying Eq. (5.15) with  $x = \phi$  and  $X_{\setminus x} = \text{mb}(x) = \{\pi, \mathbf{c}\}$

$$q(\pi) \propto \exp \left( \int \prod_n q(c_n) \left( \log p(\pi; \alpha) + \sum_n \log p(c_n | \pi) \right) d\mathbf{c} \right) \quad (\text{C.1})$$

where we note that the integral w.r.t  $\mathbf{c}$  only affects the second summation term in the exponent.

Because both  $p(\pi; \alpha)$  and  $p(c_n | \pi)$  are in the exponential family, we can write,

$$q(\pi) \propto \exp \left( \eta(\alpha) t(\pi) + \eta(\pi) \sum_n -A(\eta(\alpha)) \int q(c_n) t(c_n) dc_n - N A(\eta(\pi)) \right) \quad (\text{C.2})$$

where  $\eta(\alpha)$ ,  $t(\pi)$  and  $A(\eta(\alpha))$  are the natural parameters, sufficient statistics and log-partition function of  $p(\pi; \alpha)$ . Besides,  $\eta(\pi)$ ,  $t(c_n)$  and  $A(\eta(\pi))$  are the natural parameters, sufficient statistics and log-partition function of  $p(c_n | \pi)$ .

Since  $p(\pi; \alpha)$  is a Dirichlet and  $p(c_n | \pi)$  is a Categorical, the sufficient statistics of the former  $t(\pi)$  are equal to the natural parameters of the latter  $\eta(\pi)$ , as shown in Table B.1.

Besides the log-partition function of the Categorical is 0 and that of the Dirichlet is independent of  $\pi$ . Therefore, one can simplify the equation above as,

$$q(\pi) \propto \exp \left( t(\pi) \left( \eta(\alpha) + \sum_n \int q(c_n) t(c_n) dc_n \right) \right) \quad (\text{C.3})$$

where the variational distribution  $q(\pi)$  is clearly a Dirichlet distribution with updated natural parameters  $\eta(\pi') = \eta(\alpha) + \sum_n \int q(c_n) t(c_n) dc_n$ .

Therefore, the mean-field distribution for the mixture proportions is a Dirichlet with parameters  $\pi'_k$ .

$$q(\pi) \sim \text{Dir}(\pi; \pi'_k)$$

$$\pi'_k = \alpha_\pi + \sum_{n=1}^N c'_{nk}$$

where  $c'_{nk} = \int q(c_n) \mathbb{I}(c_n = k) dc_n = \mathbb{E}_{q(c_n)}[\mathbb{I}(c_n = k)]$ .

### C.3 Topic Distributions $q(\phi_t)$

The word distribution for each topic  $t$  is Dirichlet with parameters  $\phi'_t$ .

$$q(\phi_t) \sim \text{Dir}(\phi_t; \phi'_t)$$

$$\phi'_t = \alpha_\phi + \sum_{n=1}^N w_{nm} z'_{nmt}$$

### C.4 Temporal Mean and Precision $q(\tau_k)$ , $q(\lambda_k)$

The temporal Mean and Precision distributions for each component  $k$  are Normal and Gamma with parameters  $m'_{\tau_k}$ ,  $\beta'_{\tau_k}$  and  $a'_{\lambda_k}$ ,  $b'_{\lambda_k}$  respectively.

$$q(\tau_k) \sim \text{N}(\tau_k; m'_{\tau_k}, \beta'_{\tau_k} \frac{a'_\lambda}{b'_\lambda})$$

$$m'_{\tau_k} = \frac{m_\tau \beta_\tau + \sum_{n=1}^N c'_{nk} t_n}{\beta'_{\tau_k}}$$

$$\beta'_{\tau_k} = \beta_\tau + N_{c_k}$$

$$q(\lambda_k) \sim \text{Ga}(\lambda_k; a'_{\lambda_k}, b'_{\lambda_k})$$

$$a'_{\lambda_k} = a_\lambda + \frac{N_{c_k}}{2}$$

$$b'_{\lambda_k} = b_\lambda + \frac{1}{2} \sum_{n=1}^N (t_n - m'_{\tau_k})^2 + \frac{\beta_\tau}{2} (m'_{\tau_k} - m_\tau)^2$$

## C.5 Spatial Mean and Precision $q(\mu_k), q(\Delta_k)$

Consider the following expressions:

$$\begin{aligned}\bar{l}_k &= \frac{1}{N_{c_k}} \sum_{n=1}^N c'_{nk} l_n \\ S_k &= \frac{1}{N_{c_k}} \sum_{n=1}^N c'_{nk} (l_n - \bar{l}_k)^T (l_n - \bar{l}_k)\end{aligned}$$

The spatial mean and variance distributions for each component  $k$  are Normal and Wishart with parameters  $m'_{\mu_k}, \beta'_{\mu_k}$  and  $\nu'_k, W'_k$ , respectively.

$$\begin{aligned}q(\mu_k) &\sim N(\mu_k; m'_{\mu_k}, \beta'_{\mu_k} \nu'_k W'_k) \\ m'_{\mu_k} &= \frac{m_\mu \beta_\mu + N_{c_k} \bar{l}_k}{\beta'_{\mu_k}} \\ \beta'_{\mu_k} &= \beta_\mu + N_{c_k} \\ q(\Delta_k) &\sim W(\Delta_k; \nu'_k, W'_k) \\ \nu'_k &= \nu_\Delta + N_{c_k} \\ W'_k &= \left( W_\Delta^{-1} + N_{c_k} S_k + \frac{\beta_\mu N_{c_k}}{\beta_\mu + N_{c_k}} (\bar{l}_k - m_\mu)^T (\bar{l}_k - m_\mu) \right)^{-1}\end{aligned}$$

## C.6 Topic proportions $q(\theta_k)$

The topic proportions for each component  $k$  are Dirichlet with parameters  $\theta'_k$ .

$$\begin{aligned}q(\theta_k) &\sim \text{Dir}(\theta_k; \theta'_k) \\ \theta'_k &= \alpha_\theta + \sum_{n=1}^N c'_{nk} \sum_{m=1}^{L_n} w_{nm} z'_{nmt}\end{aligned}$$

## C.7 Topic Assignments $q(z_{n,m})$

Consider the following expression:

$$\mathbb{E}(\log \phi_t)_v = \int_{\phi_t} q(\phi_t; \phi'_{tv}) \log \phi_t = \Psi(\phi'_{tv}) - \Psi\left(\sum_{v=1}^V \phi'_{tv}\right)$$

The topic assignments distribution is a Categorical with parameters  $z'_{n,m,t}$ .

$$\begin{aligned} q(z_{nm}) &\sim \text{Cat}(z_{nm}; z'_{nmt}) \\ z'_{nmt} &= \frac{\tilde{z}'_{nmt}}{\sum_{t=1}^T \tilde{z}'_{nmt}} \\ \tilde{z}'_{nmt} &\propto \exp \left( \mathbb{E}(\log \phi_t)_m + \sum_{k=1}^K c'_{nk} \mathbb{E}(\log \theta_k)_t \right) \end{aligned}$$

## C.8 Component Assignments $q(c_n)$

Consider the following expressions:

$$\begin{aligned} \mathbb{E}(\log \pi)_k &= \int_{\pi} q(\pi; \pi'_k) \log \pi = \Psi(\pi'_k) - \Psi\left(\sum_{k=1}^K \pi'_k\right) \\ \mathbb{E}(\log \theta_k)_t &= \int_{\theta_k} q(\theta_k; \theta'_{kt}) \log \theta_k = \Psi(\theta'_{kt}) - \Psi\left(\sum_{t=1}^T \theta'_{kt}\right) \end{aligned}$$

where  $\Psi(\cdot)$  corresponds to the digamma function.

The mixture assignments distribution is a Categorical with parameters  $c'_{nk}$ .

$$\begin{aligned} q(c_n) &\sim \text{Cat}(c_n; c'_{nk}) \\ c'_{nk} &= \frac{\tilde{c}'_{nk}}{\sum_{k=1}^K \tilde{c}'_{nk}} \\ \tilde{c}'_{nk} &\propto \exp \left( \mathbb{E}(\log \pi)_k + \sum_{m=1}^{M_n} w_{nm} \sum_{t=1}^T z_{nmt} \mathbb{E}(\log \theta_k)_t \right. \\ &\quad + \mathbb{I}(k = K) (\log \text{Hist}(l_n; L_B) + \log \text{Hist}(t_n; T_B)) \\ &\quad + \mathbb{I}(k \neq K) \left( -\log 2\pi + \frac{1}{2} \left( 2 \log 2 + \Psi\left(\frac{\nu'_k}{2}\right) + \Psi\left(\frac{\nu'_k - 1}{2}\right) \right. \right. \\ &\quad \left. \left. + \log |W'_k| - \frac{2}{\beta'_{\mu_k}} - \nu'_k (l_n - m'_{\mu_k})^T W_k (l_n - m'_{\mu_k}) \right) \right. \\ &\quad \left. - \frac{1}{2} \log 2\pi + \frac{1}{2} \left( \Psi(a'_k) - \log b'_k - \frac{1}{\beta'_{\tau_k}} - \frac{a'_k}{b'_k} (t_n - m'_{\tau_k})^2 \right) \right) \end{aligned}$$





## Warble Topics in “La Mercè”

### D.1 Topic Distributions in “La Mercè” 2014

- Topic 0** dice puedo pasa pedro relax buen playa gràcia follow jordi  
**Topic 1** espectacular santa sólo beer concerts beautiful pluja internet geniales heart  
**Topic 2** ciutadella ver feliz barca así da jajaja ayer plaza h  
**Topic 3** montjuic end fuegos palabras país seguir iphone joder two  
**Topic 4** party one publicar photo crappy friends vols vol bonita pot  
**Topic 5** museu macba contemporani fan veient negra visca moment millor gallina  
**Topic 6** vaya mola sants coming nonono j nice original red ferran  
**Topic 7** felicidades ir km celebrar pp güell amor saben par p  
**Topic 8** catedral born igersbcn deja autumn momento fent impressionant caves mnac  
**Topic 9** bon mostracat espero especial cuenta pronto ole perfecte despierta tope  
**Topic 10** pues liampayne horas music home parte farem bracafé beauty montaña  
**Topic 11** messi rambla cosas personas ajuntament thanks club sun nova vine  
**Topic 12** montjuic w mas dels família piromusical castell fiesta amazing park  
**Topic 13** passeig dias nueva tibidabo disfrutando tiempo creo rokira piro  
**Topic 14** piromusical plaça despanya font poder part video running sueño teatre  
**Topic 15** sant barceloneta x felicitats back youre ever govern people che  
**Topic 16** dart manera friends apolo ciudad dun vuelta cute mercè duda  
**Topic 17** platja bogatell txarango concert manel ganas dormir k mejores new  
**Topic 18** happy genial veure igers somnis cosas die año get first  
**Topic 19** maria castells dir tricentenari programa mierda festival torre quién dóna  
**Topic 20** madrid muchas go música madre metro fc hora pena vamos  
**Topic 21** gracias spain casa sagrada familia mañana time acabo fiesta años  
**Topic 22** diada ve falta felip verdad zijk sky petit grande °c  
**Topic 23** st gothic seen leer gobierno visita musical kiwi pèssima  
**Topic 24** plaça dia jaume catalunya day mejor ahora im love bueno  
**Topic 25** parc camp palau today mundo gente like dont city puede  
**Topic 26** im q gran mercé hoy nit día españa noche cat  
**Topic 27** pell debe viure avda lany alcohol cas sabe gol  
**Topic 28** sort mujeres ok alla pesado cantante verdadero turismetlímit hecho vemos

**Topic 29** vida foto festa concierto joan vez bona quiero buenas cataluña

# Useful Expectations for PFA and BPFA

## E.1 Entropy of a Multinomial RV

The entropy of a Multinomial r.v.  $x_{np:} = [x_{np1} \cdots x_{npK}]$  with  $y_{np}$  trials and  $\gamma_{np:} = [\gamma_{np1} \cdots \gamma_{npK}]$  distributed according  $Q(x_{np:}; y_{np}, \gamma_{np:})$  can be computed as follows,

$$\mathbb{H}(x_{np:}) = \mathbb{E}_Q [-\log Q(x_{np:}; y_{np}, \gamma_{np:})] \quad (\text{E.1})$$

where  $\mathbb{E}_Q$  refers to the expectation w.r.t. distribution  $Q(x_{np:}; y_{np}, \gamma_{np:})$ .

By substituting the pmf from Eq. (B.4) in the definition of entropy above, we can use the logarithm properties to transform products into sums and push the linear operator of expectation inside as follows

$$\mathbb{H}(x_{np:}) = -\log y_{np}! - \sum_{k=1}^K \mathbb{E}_Q[x_{npk}] \log \gamma_{npk} + \sum_{k=1}^K \mathbb{E}_Q[\log x_{npk}!]. \quad (\text{E.2})$$

Given that the individual components of the Multinomial random variable are distributed according to a Binomial with  $y_{np}$  trials and  $\gamma_{npk}$  probability, the expectation of the random variable  $x_{npk}$  is equal to the mean of the  $k$ -th binomial  $\mathbb{E}_Q[x_{npk}] = y_{np} \gamma_{npk}$ . Thus, we can rewrite the previous expression as,

$$\mathbb{H}(x_{np:}) = -\log y_{np}! - y_{np} \sum_{k=1}^K \gamma_{npk} \log \gamma_{npk} + \sum_{k=1}^K \mathbb{E}_Q[\log x_{npk}!]. \quad (\text{E.3})$$

Finally, we compute the expectation of the logarithm of the factorial of the random variable  $x_{npk}$ . Again, we take advantage that each component is distributed according to a binomial to express this expectation as,

$$\mathbb{E}_Q[\log x_{npk}!] = \sum_{x_{npk}=1}^{y_{np}} \binom{y_{np}}{x_{npk}} \gamma_{npk}^{x_{npk}} (1 - \gamma_{npk})^{y_{np} - x_{npk}} \log x_{npk}!. \quad (\text{E.4})$$

Putting it all together, the entropy of a Multinomial r.v. is given by the closed-form

expression,

$$\mathbb{H}(x_{np}) = -\log y_{np}! - y_{np} \sum_{k=1}^K \gamma_{npk} \log \gamma_{npk} + \sum_{k=1}^K \sum_{x_{npk}=0}^{y_{np}} \binom{y_{np}}{x_{npk}} \gamma_{npk}^{x_{npk}} (1 - \gamma_{npk})^{y_{np} - x_{npk}} \log x_{npk}! \quad (\text{E.5})$$

which has a computational cost  $\mathcal{O}(K(y_{np} + 1))$  linear in the number of trials  $y_{np}$  and the number of components  $K$ . However, the  $\mathbb{E}_Q[\log x_{npk}!]$  is not needed for the evaluation of ELBO since it cancels out with another term in the equation, as we will show later.

## E.2 Entropy of a Gamma RV

The entropy of a Gamma r.v.  $l_{nk}$  with shape  $\beta_{nk1}$  and scale  $\beta_{nk2}$  distributed according to  $Q(l_{nk}; \beta_{nk1}, \beta_{nk2})$  can be computed as follows,

$$\mathbb{H}(l_{nk}) = \mathbb{E}_Q[-\log Q(l_{nk}; \beta_{nk1}, \beta_{nk2})] \quad (\text{E.6})$$

where  $\mathbb{E}_Q$  refers to the expectation w.r.t. distribution  $Q(l_{nk}; \beta_{nk1}, \beta_{nk2})$ .

By substituting the pdf from Eq. (B.9) in the definition above, we can apply the product and power rules of logarithms, and push the expectation operator, which is linear, inside as follows,

$$\mathbb{H}(l_{nk}) = \log \Gamma(\beta_{nk1}) + \beta_{nk1} \log \beta_{nk2} - (\beta_{nk1} - 1) \mathbb{E}_Q[\log l_{nk}] + \mathbb{E}_Q[l_{nk}] / \beta_{nk2}. \quad (\text{E.7})$$

To compute the  $\mathbb{E}_Q[l_{nk}]$  and  $\mathbb{E}_Q[\log l_{nk}]$ , we can take advantage that the Gamma distribution is part of the Exponential Family and hence, the expectation of its sufficient statistic are the partial derivatives of its cumulant. By considering the natural parameterization of the Gamma distribution in Table B.1, one can compute the partial derivatives of the cumulant  $A(\boldsymbol{\eta}) = \log \Gamma(\eta_1 + 1) - (\eta_1 + 1) \log(-\eta_2)$  as,

$$\mathbb{E}_Q[\log l_{nk}] = \frac{\delta A(\eta_1, \eta_2)}{\delta \eta_1} = \Psi(\eta_1 + 1) - \log(-\eta_2) \quad (\text{E.8})$$

$$\mathbb{E}_Q[l_{nk}] = \frac{\delta A(\eta_1, \eta_2)}{\delta \eta_2} = \frac{\eta_1 + 1}{-\eta_2} \quad (\text{E.9})$$

where  $\eta_1 = \beta_{nk1} - 1$  and  $\eta_2 = \frac{-1}{\beta_{nk2}}$  and hence, the expectations in the original parametrization are,

$$\mathbb{E}_Q[\log l_{nk}] = \frac{\delta A(\eta_1, \eta_2)}{\delta \eta_1} = \Psi(\beta_{nk1}) + \log(\beta_{nk2}) \quad (\text{E.10})$$

$$\mathbb{E}_Q[l_{nk}] = \frac{\delta A(\eta_1, \eta_2)}{\delta \eta_2} = \beta_{nk1} \beta_{nk2}. \quad (\text{E.11})$$

Putting it all together, the entropy of a Gamma r.v. is given by the expression,

$$\mathbb{H}(l_{nk}) = \beta_{nk1} + \log \Gamma(\beta_{nk1}) + \log(\beta_{nk2}) + (1 - \beta_{nk1}) \Psi(\beta_{nk1}). \quad (\text{E.12})$$

### E.3 Expectation of the log of a Poisson - PFA

Here, we develop the expectation of the logarithm of a Poisson distribution  $p(x_{npk}; l_{nk}\phi_{kp})$  w.r.t. the mean-field distribution  $Q_L(x_{np}; l_{nk}; \gamma_{npk}^L, \beta_{nk1}^L, \beta_{nk2}^L)$ , which appears in Section 8.2.1. We start by applying logarithm properties to the Poisson pdf in Eq. (B.5) and push the expectation operator inside as follows,

$$\mathbb{E}_Q[\log p(x_{npk}; l_{nk}\phi_{kp})] = \mathbb{E}_Q[x_{npk} \log l_{nk}] - (1 - \log \phi_{kp})\mathbb{E}_Q[x_{npk}] - \mathbb{E}_Q[\log x_{npk}!] \quad (\text{E.13})$$

where  $\mathbb{E}_Q$  refers to the expectation w.r.t.  $Q_L(x_{np}; l_{nk}; \gamma_{npk}^L, \beta_{nk1}^L, \beta_{nk2}^L)$ . Next, given that  $Q$  is a factorized distribution, we can compute the expectation of each term independently as,

$$\mathbb{E}_Q[x_{npk} \log l_{nk}] = \mathbb{E}_{Q(x_{np}; y_{np}, \gamma_{np})}[x_{npk}] \mathbb{E}_{Q(l_{nk}; \beta_{nk1}, \beta_{nk2})}[\log l_{nk}] \quad (\text{E.14})$$

$$\mathbb{E}_Q[x_{npk}] = \mathbb{E}_{Q(x_{np}; y_{np}, \gamma_{np})}[x_{npk}] \quad (\text{E.15})$$

$$\mathbb{E}_Q[\log x_{npk}!] = \mathbb{E}_{Q(x_{np}; y_{np}, \gamma_{np})}[\log x_{npk}!] \quad (\text{E.16})$$

where each of these expectations has been calculated in the previous section for the entropy of a Multinomial and Gamma random variable. Note that, here the expectations  $\mathbb{E}_{Q(x_{np}; y_{np}, \gamma_{np})}$  are taken w.r.t. the Multinomial distribution and  $\mathbb{E}_{Q(l_{nk}; \beta_{nk1}, \beta_{nk2})}$  w.r.t the Gamma distribution.

Putting it all together, the expectation is given by the expression,

$$\begin{aligned} \mathbb{E}_Q[\log p(x_{npk}; l_{nk}\phi_{kp})] &= y_{np}\gamma_{npk}(\Psi(\beta_{nk1}) + \log(\beta_{nk2}) + \log \phi_{kp} - 1) \\ &\quad - \sum_{x_{npk}=1}^{y_{np}} \binom{y_{np}}{x_{npk}} \gamma_{npk}^{x_{npk}} (1 - \gamma_{npk})^{y_{np}-x_{npk}} \log x_{npk}!. \end{aligned} \quad (\text{E.17})$$

Note that the term  $\mathbb{E}_Q[\log x_{npk}!]$  is not required to compute for the evaluation of the ELBO because it cancels out with the same term in the entropy.

### E.4 Expectation of the log of a Gamma

Similar to the entropy of a gamma random variable, we now develop the expectation of the logarithm of a Gamma Distribution w.r.t. a different Gamma distribution  $Q_L(l_{nk}; \beta_{nk1}^L, \beta_{nk2}^L)$ , which also appears in Section 8.2.1.

We start by applying the logarithm properties to the Gamma pdf in Eq. (B.9) and pushing the expectation operator inside,

$$\mathbb{E}_Q[\log p(l_{nk}|r_k, p_k)] = -\log \Gamma(r_k) - r_k \log p_k + (r_k - 1)\mathbb{E}_Q[\log l_{nk}] - \mathbb{E}_Q[l_{nk}]/p_k. \quad (\text{E.18})$$

By substituting the expectations above with those from Eqs. (E.11) (E.10), we can express this expectation as,

$$\mathbb{E}_Q[\log p(l_{nk}|r_k, p_k)] = -\log \Gamma(r_k) - r_k \log p_k + (r_k - 1)(\Psi(\beta_{nk1}) + \log(\beta_{nk2})) - \frac{\beta_{nk1}\beta_{nk2}}{p_k}. \quad (\text{E.19})$$

## E.5 Entropy of Jointly Zero Truncated Poisson RVs

The entropy of a  $K$  Poisson random variables with  $K$  rates given by  $\xi_{np\cdot}$  which cannot be 0 all at the same time can be computed as follows,

$$\mathbb{H}(x_{np\cdot}) = \mathbb{E}_Q [-\log Q(x_{np\cdot}; \xi_{np\cdot})] \quad (\text{E.20})$$

where  $\mathbb{E}_Q$  refers to the expectation w.r.t. distribution  $Q(x_{np\cdot}; \xi_{np\cdot})$ .

By substituting the pmf  $p(x_{np\cdot}; \xi_{np\cdot})$  given by Eq. (8.56) in the definition of entropy above, we can use the logarithm properties to transform products into sums and push the linear operator of expectation inside as follows,

$$\mathbb{H}(y_{np\cdot}) = \log \left( e^{\sum_{k=1}^K \xi_{npk}} - 1 \right) - \sum_{k=1}^K \mathbb{E}_Q [x_{npk}] \log \xi_{npk} + \mathbb{E}_Q [\log x_{npk}!]. \quad (\text{E.21})$$

Thanks that this distribution is in the exponential family, the first expectation term can be derived as follows,

$$\mathbb{E}_Q [x_{npk}] = \mathbb{E}_Q [t(x_{npk})] = \frac{\delta A(\eta(\xi_{npk}))}{\delta \eta(\xi_{npk})} = \frac{\xi_{npk}}{1 - e^{-\sum_{k=1}^K \xi_{npk}}}. \quad (\text{E.22})$$

Finally, the expectation term  $\mathbb{E}_Q [\log x_{npk}!]$  is given by the sum,

$$\mathbb{E}_Q [\log x_{npk}!] = \sum_{x_{np\cdot} \in \{\mathbb{Z}^K | x_{np\cdot} \neq [0 \dots 0]\}} \log x_{npk}! p(x_{np\cdot}; \xi_{np\cdot}) \quad (\text{E.23})$$

where the summation set are the vectors of  $K$  non-negative integers which are not all 0 at the same time. To the best of our knowledge a close-form expression does not exist to compute this entropy in finite time. Fortunately, this term will cancel out with an expectation term in the bound, so we can avoid its computation. In conclusion, the entropy of the jointly zero truncated Poisson required for the lower bound is given by,

$$\mathbb{H}(y_{np\cdot}) = \log \left( e^{\sum_{k=1}^K \xi_{npk}} - 1 \right) - \frac{1}{1 - e^{-\sum_{k=1}^K \xi_{npk}}} \sum_{k=1}^K \xi_{npk} \log \xi_{npk}. \quad (\text{E.24})$$

## E.6 Expectation of the log of a Poisson - BPFA

The expectation of the logarithm of the Poisson distribution that appears in the ELBO in Section 8.3.1 can be derived as follows,

$$\mathbb{E}_Q [\log p(x_{npk}; l_{nk} \phi_{kp})] = \mathbb{E}_Q [x_{npk} \log l_{nk}] - (1 - \log \phi_{kp}) \mathbb{E}_Q [x_{npk}] - \mathbb{E}_Q [\log x_{npk}!] \quad (\text{E.25})$$

where  $\mathbb{E}_Q$  refers to the expectation w.r.t.  $Q_L(x_{np\cdot}, l_{nk}; \xi_{npk}^L, \beta_{nk1}^L, \beta_{nk2}^L)$ . Again, we can compute the expectation for each term separately because  $Q$  is a factorized distribution.

The first expectation can be split in two as follows,

$$\mathbb{E}_Q [x_{npk} \log l_{nk}] = \mathbb{E}_{Q(x_{np\cdot}; \xi_{np\cdot})} [x_{npk}] \mathbb{E}_{Q(l_{nk}; \beta_{nk1}, \beta_{nk2})} [\log l_{nk}] \quad (\text{E.26})$$

where the first expectation is given by Eq. (E.22) and the second, by Eq. (E.10).

The rest of expectations were also given by Eq. (E.22) and Eq. (E.23), respectively, but fortunately,  $\mathbb{E}_Q[\log x_{npk}!]$  appears with the sign reversed and it cancels out with the term in the entropy.

Therefore, the contribution of this expectation to the overall lower bound can be written as,

$$\mathbb{E}_Q[\log p(x_{npk}; l_{nk}\phi_{kp})] = \frac{\xi_{npk}}{1 - e^{-\sum_{k=1}^K \xi_{npk}}} (\Psi(\beta_{nk1}) + \log(\beta_{nk2}) + \log \phi_{kp} - 1). \quad (\text{E.27})$$