Concept Discovery and Argument Bundles in the Experience Web

Xavier Ferrer[1,2] and Enric Plaza[1]

[1]Artificial Intelligence Research Institute (IIIA), Spanish National Research Council (CSIC), Campus U.A.B., Bellaterra, Catalonia (Spain)
[2]Universitat Autònoma de Barcelona, Bellaterra, Catalonia (Spain)
{xferrer,enric}@iiia.csic.es, www.iiia.csic.es

Abstract. In this paper we focus on a particular interesting web user-generated content: people's experiences. We extend our previous work on aspect extraction and sentiment analysis and propose a novel approach to create a vocabulary of basic level concepts with the appropriate granularity to characterize a set of products. This concept vocabulary is created by analyzing the usage of the aspects over a set of reviews, and allows us to find those features with a clear positive and negative polarity to create the bundles of arguments. The argument bundles allow us to define a concept-wise satisfaction degree of a user query over a set of bundles using the notion of fuzzy implication, allowing the reuse experiences of other people to the needs a specific user.

Keywords: experience web, sentiment analysis, arguments, aspect extraction, basic level concepts

1 Introduction

Our work is developed in the framework of the the Experience Web [9]. This framework proposed to enlarge the paradigm of Case-based Reasoning (CBR), based on solving new problems by learning from past experiences, and include all forms of experiences about the real world expressed in the web as user-contributed content. The final goal is to *reuse* this collective experience in helping new individuals (the "users") in taking a more informed decision according to their preferences, which can be different from the preferences of the individuals who have expressed their experiences on the web. Relating these two extreme points, from numerous but varied individual experiences to a specific user request, is the overall goal of Experience Web approach, and this paper presents a complete instance of the approach.

In this approach, we focus on praxis and usage, and we want to analyze how users express their experiences about daily life; in this paper we will focus on the usage of digital cameras. A main goal is to discover the vocabulary they use, which need not be the same as the classical feature list describing the different aspects of a camera (e.g. 4GB RAM). Our goal is to use this vocabulary to elucidate the main pros and cons of each camera, according to the user reviews.

To this end, we analyze textual reviews of user experiences with digital cameras and identify the set of aspects the users use and the polarity of the sentiment words associated with them [13,14]. Aspects are grouped in basic level concepts, creating a new concept vocabulary, to overcome the disparate granularity of the extracted aspects. Those concepts with a strong positive polarity over the set of reviews of a product are considered pros, while those with a strong negative polarity are considered cons.

We call a bundle of arguments the set of main pros and cons of a camera. We take this approach, already envisioned in [9], because the pros and cons allows us to reuse the knowledge for other users with other individual preferences. To support this reuse, we introduce the notion of query satisfaction by a bundle of arguments. The query expresses a new individual knowledge about her preferences (e.g. she's a travel photographer and needs long battery life).

The paper is organized as follows: section 2 describes the discovery of basic level concepts from user reviews. Next in sections 3 and 4 we present the three different types of argument bundles and define a user query. Evaluation results are presented in section 5, followed by related research in section 6, and conclusions in section 7.

2 Aspects and Basic Level Concepts

In our previous work on social recommender systems we harnessed knowledge from product reviews, and characterized every product by a set of aspect-sentiment pairs extracted from its reviews [13]. Based on these characterizations, we ranked and selected the most useful aspects for recommendation [14]. However, even after identifying the most useful aspects for recommendation, we still processed synonymous aspects and aspects referencing the same concept (such as *sensor* and *cmos*) as different aspects, adding noise to the recommendation process.

In this work, we use a similar approach to [13] in order to extract the set of salient aspects used to define important characteristics of photographic digital cameras. We call *aspect vocabulary* $\mathcal{A}$ the set of extracted aspects. However, instead of characterizing the products directly by the aspect vocabulary, we group them in *basic level concepts*. According to Rosch et al. [10], basic level concepts (BLC) are those that strike a tradeoff between two conflicting principles of conceptualization: inclusiveness and discrimination. They found that there is a level of inclusiveness that is optimal for human beings in terms of providing optimum cognitive economy. This level of inclusiveness is called the basic level, and concept or categories at this level are called basic-level concepts.

Research in the field of identifying basic level concepts is mostly oriented to improve the *word sense disambiguation* task. For instance, the class-based word sense disambiguation [6] approach requires to mark words by hand in a corpus as pertaining to one semantic class, that is interpreted as one BLC. Once the corpus is marked, several supervised classifiers are trained to assign the proper semantic class to each ambiguous word. In our approach, we create a collection of basic level concepts in an unsupervised way from the review corpus,

where each BLC assembles a set of aspects that, according to our analysis, are used in a similar way by the reviewers. As we show in section 2.1, we estimate this similarity by taking into account semantic similarity and evaluating the coherence/incoherence of the sentiment values of the aspects assembled in a given BLC. Synonymy is a special case of aspects being semantically equivalent.

Consider, for instance, these three aspects in $\mathcal{A}$: *picture*, *pic* and *jpeg*. One may surmise people using those words in reviews are in fact referring to the same basic level concept, i.e. *the picture obtained by my digital camera*. Thus, we could consider that different reviews in the corpus using those words are referring to the same BLC, because they have the same intended meaning.

In this section we present a method to create a concept vocabulary $\mathcal{C}$ formed by a collection of BLCs. This concept vocabulary is useful to practically reuse other people's experiences with digital cameras because it abstracts the concrete terms used in the corpus as given by the aspect extraction approach. The creation of a collection of basic level concepts consist of three steps: 1) identifying synonymous aspects, 2) building a hierarchical clustering using the semantic, syntactic and sentiment similarities among aspects, and 3) creating a concept vocabulary $\mathcal{C}$ of basic level concepts from the hierarchical clusters.

2.1 Hierarchical Clustering of Aspects

The first step is to identify the synonyms of the aspects in the aspect vocabulary $\mathcal{A}$ using WordNet, a lexical database of English. Every aspect a in $\mathcal{A}$ is mapped to the corresponding WordNet *synset* with the same noun word form, if it exists, and is disambiguated by identifying the synset with the shortest aggregated WordNet *Path Distance* [7] to a set of manually selected WordNet synsets formed by the top 5 most frequent aspects of the aspect vocabulary. The aspects that have a synonymy relation among them are grouped together into *aspect groups* G_j. Aspects without synonyms form a group of cardinality 1.

Next, we iteratively cluster the most similar groups of aspects and create a dendrogram. The set of basic level concepts will be selected from that dendogram. To cluster the aspect groups we use an unsupervised bottom-up hierarchical clustering algorithm that takes the most similar pair of groups at each stage and puts them together in a higher level group. We will define now similarity measures over aspects and over groups. The similarity measure between two aspects is:

$$Sim_A(a_i, a_j) = \alpha \cdot \Gamma(a_i, a_j) + \beta \cdot \Phi(a_i, a_j) + \gamma \cdot \Lambda(a_i, a_j)$$

where α, β and γ are weighting parameters in $[0, 1]$ such that $\alpha + \beta + \gamma = 1$. The values of Sim_A are in $[0, 1]$. Functions $\Gamma(a_i, a_j)$, $\Phi(a_i, a_j)$ and $\Lambda(a_i, a_j)$ estimate aspect similarity in three different dimensions:

- Semantic Similarity (Γ): Compares two aspect co-occurrence vectors to estimate the similarity between aspects [11].
- String Similarity (Φ): Uses the Jaro-Winkler distance to estimate the string similarity between two aspects.

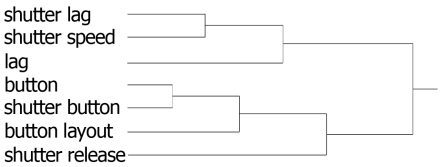

Fig. 1. Part of the dendrogram showing the clustering of concept *button*.

– PhotoDict (Λ): *PhotoDict* is a small taxonomy of camera-related terms, where similarity is measured as the shortest path between two terms. The taxonomy is automatically generated from a camera related vocabulary existing in the Web, but its creation is out of the scope of this paper.

The similarity Sim_G between two groups of aspects G_i and G_j is defined as:

$$Sim_G(G_i, G_j) = \frac{1}{|G_i||G_j|} \sum_{n=1}^{|G_i|} \sum_{m=1}^{|G_j|} Sim_A(a_n, a_m)$$

There is a special treatment of compound nouns in clustering. Since compound nouns are formed by two or more words (e.g. *image quality*), we group them with the most frequent aspect among the words forming the compound. The result of the hierarchical clustering is a dendrogram (or clustering tree) of aspects; Figure 1 shows a small part of the resulting dendrogram for concept *button*. Since hierarchical clustering gives multiple partitions (clusterings) at different levels, next we have to select one partition to create our concept vocabulary.

2.2 Concept and Vocabulary Creation

We are interested in selecting a partition from the hierarchical clustering dendrogram that is able to describe the basic level concepts of digital cameras based on the user experiences of our corpus. The groups of aspects forming the selected partition will become our concept vocabulary $\mathcal{C}$.

To select the best partition, we cut the dendrogram at different levels. Then, for each partition, we analyze the coherence degree of the sentiment values in each aspect group. If the sentiments of the aspects of a group G cohere into a clear positive, negative, or neutral value, we consider G a potential basic level concept. For instance, let *picture*, *photo* and *image* be three aspects in a group. If those three aspects are used by people to refer to the same concept ('picture obtained by my digital camera'), then the sentiment values of those aspects with respect to the reviews of each product should have a high coherence degree.

The *Partition Ranking* score $R(K)$ of a partition K is estimated as follows:

$$R(K) = \frac{1}{|K|} \sum_{i=1}^{|K|} IS(G_i)$$

Concept Name	Aspects in Concept
Storage	storage, capacity, sd card, sdhc card, cf card
Button	lag, shutter release, shutter speed, shutter lag, shutter button, button, button layout
Battery	battery, battery life, battery pack

Table 1. Three of the basic level concepts in $\mathcal{C}$ and their aspects.

where $|K|$ is the number aspect groups that form the partition K. The coherence degree is estimated by $IS(G_i)$, the average sentiment similarity among the aspects in a group G_i. The higher *R(K)*, the better the partition K.

The average sentiment similarity *IS* of a group of aspects G is the average cosine similarity among all pairs of aspects in G:

$$IS(G) = \frac{1}{|G| \cdot (|G| - 1)} \sum_{i=1}^{|G|} \sum_{j=1, j \neq i}^{|G|} cos(D(a_i), D(a_j))$$

where $cos(D(a_i), D(a_j))$ is the cosine of the angle between aspect vectors $D(a_i)$ and $D(a_j)$. An aspect vector is $D(a) = (S_{av}(p_i, a))_{i \in 1,\ldots,|\mathcal{P}|}$, where $S_{av}(p_i, a) \in [0, 1]$ is defined as the normalized sentiment average over the set of sentences from the reviews of product p_i in which aspect a occurs.

In our experiments, we only considered partitions with 30 to 40 groups, a reasonable concept vocabulary size for our purposes. The partition K with 36 groups, that had the highest $R(K)$, was selected. Each group of aspects is considered a basic level concept (BLC) and these 36 BLCs form the concept vocabulary $\mathcal{C}$. We will use $\mathcal{C}$ in Section 3 to create the bundles of arguments. Table 1 presents a small example of 3 concepts in $\mathcal{C}$ and their aspects. The *concept name* column corresponds to the most frequent aspect of each concept.

3 Bundle of Arguments

In this Section we characterize the set of products $p \in \mathcal{P}$ based on the concept vocabulary $\mathcal{C}$ created in previous section. Let $p \in \mathcal{P}$ be a product, $C \in \mathcal{C}$ a concept, and $Occ(p, C)$ the set of sentences from the reviews of product p in which any of the aspects that form the concept C appears. By analyzing the sentiment values of $Occ(p, C)$, we infer whether the people's experiences about a concept C of a product p have a positive or negative overall sentiment. If the overall polarity of the occurrences of a concept over the reviews of a product is positive, we consider that concept to be a *pro* argument for the product. If the overall polarity is negative, we consider that concept a *con* argument for the product. Finally, if the overall polarity of the occurrences of a concept over the reviews of a product is not clearly positive or negative, we consider the concept a *moot* argument of the product. By considering the pros, cons and moots of a product over the set of concepts in the concept vocabulary, we obtain a characterization about what people like or dislike of that product. The union of the pro, con, and

moot arguments, considering all concepts in the concept vocabulary $\mathcal{C}$, form the bundle of arguments B of a product p: $B(p) = \mathit{Pros}(p) \cup \mathit{Cons}(p) \cup \mathit{Moots}(p)$.

Let $\mathit{Args}(p) = \{\mathit{Arg}_i\}_{i=1,\ldots,|\mathcal{C}|}$ be the arguments of a product p, and let $\mathit{Arg} = \langle p, C, \mathbf{s}\rangle$ be an argument formed by a tuple of a product $p \in \mathcal{P}$, a concept $C \in \mathcal{C}$ and an aggregated sentiment $\mathbf{s}$ (calculated by aggregating the sentiment values of $\mathit{Occ}(p, C)$, to be defined later). The *Pros*, *Cons* and *Moots* are defined:

$$\begin{aligned}\mathit{Pros}(p) &= \{\mathit{Arg} \in \mathit{Args}(p) | \mathit{Arg}.\mathbf{s} > \delta\ \}\\ \mathit{Cons}(p) &= \{\mathit{Arg} \in \mathit{Args}(p) | \mathit{Arg}.\mathbf{s} < -\delta\ \}\\ \mathit{Moots}(p) &= \{\mathit{Arg} \in \mathit{Args}(p) | -\delta \leq \mathit{Arg}.\mathbf{s} \leq \delta\}\end{aligned}$$

where δ is a threshold that determines when an argument is considered *Pro*, *Con* or *Moot*; we will show later how δ depends on the bundle type $(\delta_G, \delta_\sigma, \delta_F)$.

In this work we consider three different methods to create a bundle of arguments: Gini (B_G), Agreement (B_σ), and Cardinality (B_F) bundles. Each bundle type is built by a different sentiment aggregation measure; moreover, they share a parameter Δ that considers moot those arguments with a very small $\mathit{Occ}(p, C)$. We will now define the three types of argument bundles: B_G, B_σ and B_F.

Gini Bundle (B_G): An argument in B_G has the form $\langle p, C, S_G(p, C)\rangle$, where the polarity value S_G is calculated using the average sentiment $S_{av}(p, C)$ and then using the *Gini Coefficient* [15] to penalize the average sentiment according to the degree of dispersion of sentiment values: $S(p, C) = S_{av}(p, C)(1 - \mathit{Gini}(p, C))$.

$$S_G(p, C) = \begin{cases} 0 & \text{if } |\mathit{Occ}(p, C)| < \Delta \text{ or } -\delta_G > S(p, C) < \delta_G \\ S(p, C) & \text{otherwise} \end{cases}$$

Notice that, when $|\mathit{Occ}(p, C)| < \Delta$, we consider that we don't have enough reviews of product p with concept C and we assign a neutral sentiment value. Similarly, when $-\delta_G < S_{av}(p, C) \cdot (1 - \mathit{Gini}(p, C)) < \delta_G$, we consider that the polarity is not strong enough to define an argument as a pro or a con, and we assign a neutral sentiment value. Finally, the parameter δ_G (set to 0.1 in the experiments) determines when the argument is considered pro, con or moot.

Agreement Bundle (B_σ): Let $\mathit{Dev}(p, C)$ be the standard deviation of the sentiment values of $\mathit{Occ}(p, C)$. The agreement sentiment measure $S_\sigma(p, C)$ is the sentiment average of the sentiment values of the sentences in $\mathit{Occ}(p, C)$, for those concepts whose $\mathit{Dev}(p, C) < \delta_{max}$. This measure uses two threshold parameters δ_{max} and δ_σ. First, δ_{max} specifies the maximum acceptable standard deviation over the distribution of sentiment values in $\mathit{Occ}(p, C)$: when $\mathit{Dev}(p, C) > \delta_{max}$ we consider that we have no grounds for an informed decision on the overall polarity of C with respect to product p. Second, δ_σ specifies the threshold for an argument sentiment value to be considered a pro, a con, or a moot argument. An argument in B_σ has the form $\langle p, C, S_\sigma(p, C)\rangle$ where S_σ is defined as follows:

$$S_\sigma(p, C) = \begin{cases} 0, & \text{if } \mathit{Dev}(p, C) > \delta_{max} \text{ or } |\mathit{Occ}(p, C)| < \Delta \\ S_{av}(p, C), & \text{otherwise} \end{cases}$$

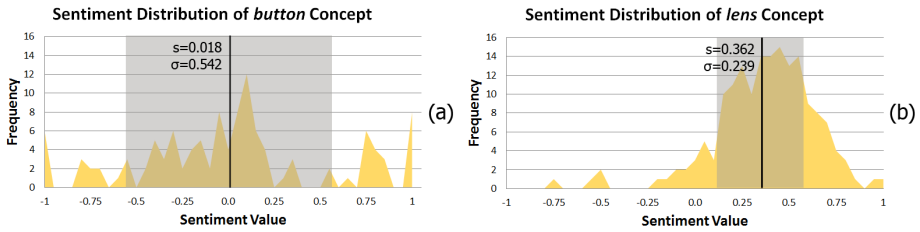

Fig. 2. Sentiment value distribution (a) *button* concept and (b) *lens* concept for Pentax K-5. Notice that values have a higher degree of dispersion in (a) than in (b).

Parameter δ_σ value is set to 0.1 in the experiments.

Figure 2 presents the sentiment value distribution of two arguments of Pentax K-5, *button (a)* and *lens (b)*. The *button* argument of the Pentax K-5 has a sentiment value deviation $\sigma = 0.542$, showing a high dispersion of sentiment values for concept *button* among the reviews of Pentax K-5. Since the deviation of the sentiment values of *button* is higher than δ_{max}, we have no clear overall polarity. On the other hand, the deviation of the sentiment values of *lens* is lower than the threshold and has a positive average sentiment ($0.235 > \delta_\sigma$). Therefore, argument *lens* is considered a pro argument with respect to Pentax K-5.

Cardinality Bundle (B_F): The cardinality bundle is created by comparing the number of positive versus negative occurrences of a concept C in $Occ(p, C)$. The number of positive (O^+) and negative (O^-) occurrences of a concept C in the reviews of a product p are defined as $O^+(p, C) = |\{x \in Occ(p, C) \,|\, \mathbf{s}(C, x) > 0\}|$ and $O^-(p, C) = |\{x \in Occ(p, C) \,|\, \mathbf{s}(C, x) < 0\}|$, where $\mathbf{s}(C, x)$ is the sentiment value in $[-1, 1]$ of concept C in sentence x.

An argument in B_F has the form $\langle p, C, S_F(p, C)\rangle$ where S_F is:

$$S_F(p,C) = \begin{cases} 0, & \text{if } \left(2 \cdot \frac{O^+}{O^+ + O^-}\right) - 1 = 0 \text{ or } |Occ(p,C)| < \Delta \\ \left(2 \cdot \frac{O^+}{O^+ + O^-}\right) - 1, & \text{otherwise} \end{cases}$$

where $O^+ = O^+(p, C)$ and $O^- = O^-(p, C)$. Notice that $S_F(p, C)$ takes values on $(0, 1]$ if $O^+ > O^-$, and in $[-1, 0)$ if $O^+ < O^-$. In the experiments we set $\delta_F = 0$ as the threshold that determines if an argument is pro, con or moot.

As a final step, we create three collections of bundles (one for each bundle type) considering the whole set of products and rescale the sentiment values of the arguments that form the bundles of the collection in a way that the most positive argument sentiment about a concept has a sentiment 1, and the most negative a sentiment -1. We rescale the rest of the sentiment values accordingly. This way, considering a collection of product bundles, the product with the best sentiment over a concept has a sentiment value of 1. When all arguments of a bundle B are rescaled we call it a normalized bundle $\overline{B}$.

4 User Query over Product Bundles

A user query defines the requirements of a user expressed using the concept vocabulary $\mathcal{C}$. Since not all requirements are equally important for the user, every requirement over a concept has a utility value. Given a set of products characterized with the normalized bundles of arguments $\overline{B}(p)$, we can decide which is the product that has a higher level of query satisfaction.

We define a *user query* $Q = \{(C_j, U(C_j))\}_{j=1,\ldots,k}$ and $k \leq |\mathcal{C}|$ as a set of concept utility pairs. Each concept utility pair $(C_j, U(C_j))$ expresses a requirement from the user over concept C_j with a utility degree $U(C_j) \in [0.5, 1]$. For instance in a query $Q = \{(lens, 0.9), (video, 0.6)\}$, the user requires high quality lens and video, although the quality of the lens is more important than the quality of the video. Furthermore, a good lens or video are more important for the user than any other feature the camera could possess.

We will now define the degree of *Query Satisfaction*, $DS(Q, \overline{B})$), that determines the degree in which a normalized bundle $\overline{B}$ satisfies a user query Q. Since t-norms and implications in fuzzy logic are defined in the interval $[0, 1]$, we need to rescale the sentiment values of all arguments that form all product bundles from [-1,1] to [0,1] by applying the linear mapping $f(\mathbf{s}) = \frac{\mathbf{s}+1}{2}$. For example, consider an argument $\langle p, lens, 0.83\rangle \in \overline{B}(p)$, the sentiment of the argument will be $f(0.83) = 0.915$. Notice that the neutral value 0 in [-1,1] is mapped to the neutral value 0,5 in [0,1].

We will first define a concept-wise satisfaction degree using the notion of fuzzy implication, specifically we will use fuzzy implication associated to the t-norm product ($\Rightarrow_{\otimes}$).

$$U(C_j) \Rightarrow_{\otimes} \mathbf{s}_j = \begin{cases} 1, & \text{if } U(C_j) \leq \mathbf{s}_j \\ \frac{\mathbf{s}_j}{U(\mathbf{s}_j)} & \text{otherwise} \end{cases}$$

where $\mathbf{s}_j$ is the rescaled sentiment value of argument $\langle p, C_j, \mathbf{s}_j\rangle$. We need now to aggregate these k concept-wise satisfaction degrees into an overall degree of bundle satisfaction of a query Q. For this purpose, we will use the t-norm product as follows:

$$DS(Q, \overline{B}(p)) = \prod_{j=1}^{k} (\, U(C_j) \Rightarrow_{\otimes} \ \mathbf{s}_j)$$

where $\mathbf{s}_j$ is the rescaled sentiment value of argument $\langle p, C_j, \mathbf{s}_j\rangle$ of the argument bundle $\overline{B}(p)$ and $\overline{B}$ is a normalized argument bundle (either $\overline{B}_G$, $\overline{B}_\sigma$ or $\overline{B}_F$).

Table 2 shows the degree of satisfaction of two user queries Q_1 and Q_2 against the cardinality bundles of two cameras: Nikon D7100 and Canon EOS70D (sentiment values are rescaled). The first query is created by a user who likes to go hiking and that is looking for a camera to capture landscape and nature while valuing fine detail. Assume her query is $Q_1 = \{(picture, 0.7), (resolution, 0.6)\}$ because she wants a camera with good image quality and resolution. Table 2 shows on the first two rows the sentiment values of the two cameras in the concepts appearing in the query. The second two rows show the satisfaction degree

Q_1 Requirements	(*picture*, 0.7)	(*resolution*, 0.6)	$DS(Q_1, \overline{B}_F)$
$\overline{B}_F$(D7100)	0.75	1.00	
$\overline{B}_F$(EOS70D)	0.97	0.50	
$U(C_j) \Rightarrow_{\otimes} s_j$ for $\overline{B}_F$(D7100)	1.00	1.00	**1.00**
$U(C_j) \Rightarrow_{\otimes} s_j$ for $\overline{B}_F$(EOS70D)	1.00	0.83	0.83

Q_2 Requirements	(*picture*, 0.7)	(*resolution*, 0.6)	(*video*, 0.9)	$DS(Q_2, \overline{B}_F)$
$\overline{B}_F$(D7100)	0.75	1.00	0.64	
$\overline{B}_F$(EOS70D)	0.97	0.50	1.00	
$U(C_j) \Rightarrow_{\otimes} s_j$ for $\overline{B}_F$(D7100)	1.00	1.00	0.72	0.72
$U(C_j) \Rightarrow_{\otimes} s_j$ for $\overline{B}_F$(EOS70D)	1.00	0.83	1.00	**0.83**

Table 2. Degree of satisfaction of two cameras for each requirement and the overall DS for the query Q_1 and the Q_2.

of the two cameras for each requirement and the overall DS for the query. Notice that satisfaction is 1 when the sentiment value is higher than the required utility value for a concept.

The second example is query $Q_2 = \{$(*picture*, 0.7), (*resolution*, 0.6), (*video*, 0.9)$\}$ (second half of Table 2) is created by a user that, besides hiking, also loves recording video. Now, according to user reviews, Canon EOS70D has an outstanding video quality (1.0), while Nikon D7100 has an average quality video (0.64). Because of this new added requirement now the higher ranking camera is Canon EOS70D instead of Nikon D7100, the best ranking camera for Q_1.

5 Evaluation

In this section we compare and evaluate the different bundles of arguments with those of DPReview.com, a renowned website specialized in digital cameras. We are keen to study the differences between the sets of pros, cons and moots of the three different bundles of arguments, B_G, B_σ and B_F, while assessing the impact that the number of reviews of a product has over the quality of the bundle of arguments. Therefore we evaluate the precision and recall of the product bundles by comparing them with the expert evaluations of products presented in DPReview. Finally, we present a ranking strategy for product bundles and compare the rankings of products obtained with each bundle type (B_G, B_σ, B_F) compared with two external product rankings (those of DPReview and Amazon).

The *Digital SLR Camera* dataset we use was extracted by us from Amazon during April 2014 [13] contained more than 20,000 user generated reviews over a set of 2,264 products. We pruned those products older than 1st January 2008 and with less than 15 user reviews, and merged any synonymous products, leaving us data on 50 products. Over the set of reviews of these products we extracted 251 different aspects, that were grouped by the hierarchical clustering algorithm presented in Section 2 into 36 concepts, that form the concept vocabulary $\mathcal{C}$. Using $\mathcal{C}$, we created three types of argument bundles for each of the 50 products as described in Section 3.

	Gini Bundle B_G	Agreement Bundle B_σ	Cardinality Bundle B_F
Avg # pros	9.42	12.44	12.65
Avg # cons	0.54	3.42	2.60
Avg # moots	26.04	20.14	20.75

Table 3. Average number of pros, cons and moot arguments for the 3 bundle types.

Comparison between Argument Bundles B_G, B_σ and B_F. Here we study the differences among the pro, con and moot arguments of the three bundle types B_G, B_σ and B_F. Since the criteria to select the arguments varies between the three bundle types, the quantity of pros, cons, and moot arguments obtained by each bundle type may differ. Table 3 presents a comparison between the average quantity of pros, cons and moot arguments of each bundle type.

The Agreement and Cardinality bundles have a similar average number of pros and cons, while Gini bundles are slightly smaller. The Gini average tends to move the argument sentiment value the towards 0 when there is dispersion in the distribution of sentiment values, and thus more arguments tend to be moots.

Next we study which concepts are considered pros in the different bundles. Figure 3 presents the quantity of pros shared between the three bundle types of each product, showing that most pros (almost 8 out of 10) are shared between two or three bundle types of a product, a good indicator of the consistency of our approach. This means that a pro concept in a B_G is also likely form part of B_σ pros and B_F pros. Furthermore, the number of pros (and also cons, not included in this figure due to lack of space) of a bundle is directly related with the number of sentences in the reviews of that product: the more reviews the more richer the bundles are. Notice that we are only studying if a concept is categorized as a pro between the 3 bundle types of a product; we are not comparing the concrete positive sentiment values of the arguments.

Bundle of Arguments Evaluation. To evaluate the quality of bundles, we compared the bundles of arguments of the 15 products with more reviews with the product pros and cons textual descriptions from DPReview. The DPReview pros and cons of a product are separately formed by lists of sentences such as 'good detail and color in JPEGs at base ISO (pro)' or 'buggy Live View / Movie Mode (con)'. In order to compare the DPReview pro and con items with our bundles of arguments, we first manually identify the concepts referenced in each item text and interpret that concept as one of the concepts in our concept vocabulary, if it exists. For instance, we consider that previous DPReview pro sentence 'good detail and color in JPEGs at base ISO' refers to the vocabulary concepts *jpeg*, *color* and *picture*, whilst 'buggy Live View / Movie Mode' refers negatively to concepts *live view* and *video*. Those sentences from DPReview that did not clearly refer to a concept in $\mathcal{C}$ were ignored. By grouping the vocabulary concepts present in the DPReview pro and con items of a product, we create the sets of DPReview pros $Pros_{dp}$ and cons $Cons_{dp}$ but without a sentiment value

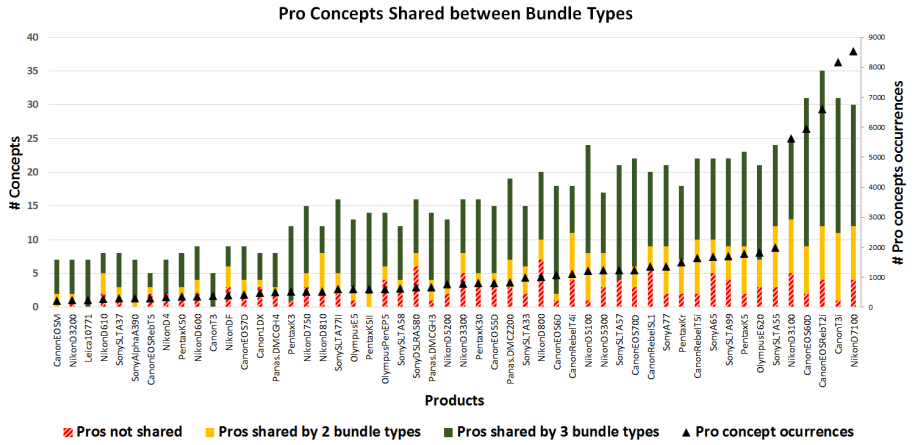

Fig. 3. Quantity of pros shared between the three bundles of arguments B_G, B_σ and B_F, together with the number of occurrences of the pro concepts in the reviews of the product.

		Precision	Recall	F_2-score	*Contradictions*
Pros	B_G	**0.567**	0.644	0.627	**0.004**
	B_σ	0.506	0.761	0.691	0.135
	B_F	0.513	**0.822**	**0.733**	0.065
Cons	B_G	0.333	0.046	0.056	0.046
	B_σ	0.285	0.558	0.468	0.132
	B_F	0.388	0.488	0.464	0.165

Table 4. Measures on precision, recall, F_2-score and contradictions between pros and cons of bundles B_G, B_σ and B_F with respect to DPReview pros and cons.

associated. We compare those DPReview sets with the pros and cons of the three different bundles of arguments of each product without taking into account the sentiment values, only whether the concept is selected as a pro or con.

Table 4 present the average precision, recall and F_2-score between the sets of pros and cons of the three bundle types and those of DPReview. We use the F_2-score to weight recall higher than precision, since we are keen to study whether the three different bundle types identify as pros and cons the same concepts listed in DPReview. Furthermore, we analyze the percentage of *contradictions*, which are those concepts selected as pros in our bundles of arguments but considered cons in DPReview and vice versa. A low percentage of contradictions is a good indicator of the quality of the bundles.

The bundle of arguments that performs best for the pro arguments is the cardinality bundle B_F, with an average recall of 0.822 and an F_2-score of 0.733. That means that the 82.2% of the concepts listed as pros of product p in DPReview also form part of the pros of the cardinality bundle $B_F(p)$. On the other hand, the sets of cons of all three bundles of arguments perform poorly. This

is because the granularity of the sentences is different between our concept vocabulary and DPReview. For us, the granularity level is given by our concept vocabulary, while DPReview sentences normally work at different levels of granularity. Furthermore, the granularity of DPReview sentences varies whether the sentence is a pro or a con. DPReview pro sentences tend to be more general: 'camera buttons and dials are useful and easily configurable', while con sentences tend to be more specific: 'the video dial is not easily accessible'. Although for us both sentences reference concept *button*, it is clear that the DPReview pro sentence better describes a general view of the buttons of the camera than the second one. Furthermore, note that the precision values of all bundles are lower than 0.6, suggesting that the sets of pros of the bundles of arguments are richer in concepts compared to those of DPReview summaries. This is not strange, since the sets of DPReview pros and cons are not exhaustive but a short list of the concepts that stand out from their point of view. The average size of the set of bundle pros is 12-14 arguments, while the average pro set size of DPReview identified concepts is 7-9. Finally, notice the low quantity of contradictions between the bundles of arguments and the DPReview sets. However low, we are interested in studying what are the most frequent concepts in contradictions.

The most common contradictions between the bundles and the set of pro and con concepts extracted from DPReview for the 15 selected products are: battery (10), viewfinder (5), recording (5) and button (3). In DPReview battery is often selected as a pro, however it is usually selected as a con in the bundles of arguments. That is because in the reviews people usually complain about the battery of a camera, while they do not seem to express positive opinions on cameras with a good battery (it would seem it is taken as a given). Other frequent contradictions are *viewfinder*, *recording* and *button*. This is because in DPReview those are commonly selected as cons for having not optimal behavior in certain types of situations (e.g. 'the video dial is not easily accessible') while the overall opinions about the rest of the buttons are positive. Therefore, our bundles will capture this average higher granularity sentiment of *button*. Similar situations are observed for *recording* and *jpeg* concepts.

Next we define the function $\Theta : B \times B \rightarrow [-1, 1]$ that estimates the degree in which a product bundle $B(p_i)$ is better or superior to another bundle $B(p_j)$:

$$\Theta(B(p_i), B(p_j)) = \frac{1}{2|\mathcal{C}|} \sum_{k=1}^{|\mathcal{C}|} \mathbf{s}_k^i - \mathbf{s}_k^j$$

where $\mathbf{s}_k^i$ and $\mathbf{s}_k^j$ are the sentiment values of respective arguments $\langle p_i, C_k, \mathbf{s}_k^i \rangle$ and $\langle p_j, C_k, \mathbf{s}_k^j \rangle$ in the bundles of products p_i and p_j. Θ is the average of these differences over all concepts in $\mathcal{C}$, a value in $[-1, 1]$. If the value of $\Theta(B(p_i), B(p_j))$ is in $(0, 1]$, then $B(p_i)$ is superior than $B(p_j)$, while if this value is in $[-1, 0)$, then $B(p_i)$ is worse than $B(p_j)$.

Using Θ, we take the 15 products with more reviews and we create a product ranking for each bundle type (B_G, B_σ and B_F). Moreover, we create two more rankings over these 15 products: 1) *DPReview Ranking*, based on the DPReview

	Spearman Rank Correlation	
Rankings	DPReview Ranking	Amazon Ranking
B_G Ranking	0.50	-0.19
B_σ Ranking	0.57	-0.33
B_F Ranking	**0.90**	0.09
Random Ranking	0.34	0.34
DPReview Ranking	1	0.33

Table 5. Spearman rank correlation of the bundle rankings with DPReview product ranking and Amazon star ratings ranking.

overall product score, and 2) *Amazon Ranking*, based on Amazon's star rating score. Whenever two or more products had the same DPReview score, such as Olympus E620 and Nikon D3100 both with a score of 72 out of 100, we only kept the product with most reviews, in this example the Nikon D3100. This left us with 9 different products. Let us now compare these rankings. The top 3 products for the B_G ranking are Nikon D7100, Pentax K-5 and SonySLT A-55. The top 3 products for B_σ are Nikon D7100, SonySLT A-99 and SonySLT A-55, and the top 3 ranked products for B_F are Nikon D7100, SonySLT A-99 and Pentax K-5. Notice that Nikon D7100 is the top product in all three bundles, and it is also ranked 1st (with a score of 85 points) in the DPReview ranking, followed by SonySLT A-99 and Pentax K-5. Table 5 shows the *Spearman Rank Correlation* of the 3 bundle rankings with the DPReview Ranking and the Amazon Ranking. We added a random ranking strategy to facilitate a baseline comparison. The random ranking correlation values were obtained by averaging the Spearman correlations of 1000 randomly generated product rankings with DPReview ranking and Amazon ranking.

The results show that B_F ranking has the highest Spearman correlation with DPReview ranking (correlation of 0.904). This value tells there is a very strong correlation between the two rankings, a good indicator of the quality of the cardinality bundles B_F. The correlations B_σ and B_G are also strong, being notably higher than the random ranking correlations. Note that the Amazon star-based ranking does not correlate with any of the bundle rankings nor the DPReview score ranking. In fact, the random ranking obtains the highest Spearman rank correlation with the Amazon star ranking, showing no strong correlation between the star-rating ranking and the bundles extracted from the reviews. This may be understandable, since two people with similar arguments about a product can give different star-rating values. Nevertheless, the fact is that Amazon's star rating cannot be used as ground truth to test the quality of the bundles.

6 Related Work

There exist numerous applications that gather knowledge from user-generated reviews, usually oriented to help other users make more informed decisions in the area of recommendation systems and CBR. The most common approach

consists in characterizing a set of products by considering product aspects (also called features) mentioned in the reviews [1,2]. In this process, the set of aspects selected to characterize a product together with the sentiment analysis of the sentences have a crucial role in the final recommendation [3,5,12]. A related work on creating BLC is [6], but they have to mark by hand a corpus with the classes (concepts) to which words belong; then they use supervised learning while we discover th BLCs in an unsupervised way.

Another focus is identifying the sets of aspects with higher positive/negative polarity to give insights into the reason why items have been chosen [8]. Those approaches need previously to group the aspects to reduce the granularity in order to provide useful recommendations, often solved by clustering aspects using background knowledge to simplify the process. Our approach is different in a sense that we create basic level concepts [10] by exploring the usage of the aspects among the user-generated reviews in an unsupervised way.

Using these basic concepts, we build the bundles of arguments by identifying the pro and con concepts over the set of reviews of a product. Finally, we define a concept-wise satisfaction degree of a user query over a set of bundles using the notion of fuzzy implication [4]. User queries are the reason we define bundles: they allow to reuse experiences of other people to the needs a specific user.

7 Conclusions and Future Work

In this paper we extend our previous work on aspect extraction and sentiment analysis and propose a method to create a vocabulary of basic level concepts with the appropriate granularity to characterize a set of cameras. This concept vocabulary is useful to practically reuse other people's experiences with digital cameras because it abstracts the concrete terms used in the corpus as given by the aspect extraction approach. By analyzing the usage of the concepts over the reviews of a product, we find those concepts that have a clearly positive or negative polarity and create the argument bundles. We present three different types of argument bundles, each one defining the pros and cons of a product based on a different criteria. The argument bundles allow us to define a satisfaction degree, interpreted in fuzzy logic and modeled with a fuzzy implication operator, between products and a user query.

An evaluation of the three types of argument bundles is performed and compared with the expert descriptions of the DPReview website, showing that the bundles of arguments correctly identify the pro and con features listed in DPReview. Moreover, the cardinality bundle ranking proved to correlate with the overall DPReview score ranking over the subset of the most frequent products, while Amazon.com star rating ranking does not correlate with neither of them.

The characterization of products by means of the bundles of arguments and BLC is promising. We have observed that the quality of a product bundle is related to the quantity of reviews of that product: the products with more reviews have a richer vocabulary of pro and con arguments, while products with fewer reviews had more moots. This can be due to two reasons that open new lines for

future work. First, improving the detection of aspects (for instance, considering also 3-gram aspects) could improve the argument bundles of those products with less reviews. And second, improving the sentiment analysis of reviews by developing a domain specific sentiment dictionary for digital cameras will enhance the accuracy of the arguments' sentiment.

Acknowledgments. This research has been partially supported by NASAID (CSIC Intramural 201550E022). We thank Lluís Godo and Pere García for their insightful comments.

References

1. L. Chen, G. Chen, and F. Wang. Recommender systems based on user reviews: the state of the art. *User Modeling and User-Adapted Interaction*, 25(2):99–154, 2015.
2. R. Dong, M. Schaal, M. O'Mahony, K. McCarthy, and B. Smyth. Mining features and sentiment from review experiences. In *Case-Based Reasoning Research and Development*, pages 59–73, 2013.
3. R. Dong, M. Schaal, M. O'Mahony, K. McCarthy, and B. Smyth. Opinionated product recommendation. In *Case-Based Reasoning Research and Development*, volume 7969 of *Lecture Notes in Computer Science*, pages 44–58. Springer, 2013.
4. P. Hájek, Ll. Godo, and F. Esteva. A complete many-valued logic with product-conjunction. *Archive for mathematical logic*, 35(3):191–208, 1996.
5. M. Hu and B. Liu. Mining opinion features in customer reviews. In *Proc. of the National Conf. on Artificial Intelligence*, pages 755–760, 2004.
6. R. Izquierdo, A. Suárez, and G. Rigau. Word vs. class-based word sense disambiguation. *J. Artif. Intell. Res. (JAIR)*, 54:83–122, 2015.
7. L. Meng, R. Huang, and J. Gu. A review of semantic similarity measures in wordnet. *Int. Journal of Hybrid Information Technology*, 6(1):1–12, 2013.
8. K. Muhammad, A. Lawlor, R. Rafter, and B. Smyth. Great explanations: Opinionated explanations for recommendations. In *Case-Based Reasoning Research and Development*, pages 244–258. Springer, 2015.
9. E. Plaza. Semantics and experience in the future web. In *Advances in Case-Based Reasoning: 9th European Conference, ECCBR 2008*, volume 5239 of *Lecture Notes in Artificial Intelligence*, pages 44–58. Springer Verlag, 2008.
10. E. Rosch. Human categorization. *Studies in cross-cultural psychology*, 1:1–49, 1977.
11. S. Sani, N. Wiratunga, S. Massie, and R. Lothian. Term similarity and weighting framework for text representation. In *Case-Based Reasoning Research and Development*, pages 304–318. Springer, 2011.
12. P. Turney. Thumbs up or thumbs down?: semantic orientation applied to unsupervised classification of reviews. In *Proc. Annual Meeting on Association for Computational Linguistics*, pages 417–424, 2002.
13. Y. Y.Chen, X. Ferrer, N. Wiratunga, and E. Plaza. Sentiment and preference guided social recommendation. In *Case-Based Reasoning Research and Development*, pages 79–94. Springer, 2014.
14. Y. Y.Chen, X. Ferrer, N. Wiratunga, and E. Plaza. Aspect selection for social recommender systems. In *Case-Based Reasoning Research and Development*, pages 60–72. Springer, 2015.
15. S. Yitzhaki. Relative deprivation and the Gini coefficient. *The Quarterly Journal of Economics*, pages 321–324, 1979.