Terminology Extraction Approaches for Product Aspect Detection in Customer Reviews

Jürgen Broß
Institute of Computer Science
Freie Universität Berlin
14195 Berlin, Germany
juergen.bross@fu-berlin.de

Heiko Ehrig
Neofonie GmbH
Robert-Koch-Platz 4
10115 Berlin, Germany
heiko.ehrig@neofonie.de

Abstract

In this paper, we address the problem of identifying relevant product aspects in a collection of online customer reviews. Being able to detect such aspects represents an important subtask of aspect-based review mining systems, which aim at automatically generating structured summaries of customer opinions. We cast the task as a terminology extraction problem and examine the utility of varying term acquisition heuristics, filtering techniques, variant aggregation methods, and relevance measures. We evaluate the different approaches on two distinct datasets (hotel and camera reviews). For the best configuration, we find significant improvements over a state-of-the-art baseline method.

1 Introduction

Identifying significant terms in a text corpus constitutes a core task in natural language processing. Fields of application are for example *glossary extraction* (Kozakov et al., 2004) or *ontology learning* (Navigli and Velardi, 2004). In this work, we particularly focus on the application scenario of *aspect-based customer review mining* (Hu and Liu, 2004; Dave et al., 2003). It is best described as a *sentiment analysis* task, where the goal is to summarize the opinions expressed in customer reviews. Typically, the problem is decomposed into three subtasks: 1) identify mentions of relevant product aspects, 2) identify sentiment expressions and determine their polarity, and 3) aggregate the sentiments for each aspect. In this paper, we only consider the first subtask, i.e., finding relevant product aspects in reviews.

More precisely, we define the problem setting as follows: Input is a homogeneous collection of customer reviews, i.e., all reviews refer to a single product type (e.g., digital cameras or hotels). The goal is to automatically derive a lexicon of the most relevant aspects related to the product type. For example, given a set of hotel reviews, we want to determine aspects such as “room size”, “front desk staff” “sleep quality”, and so on. In general, product aspects may occur as *nominal* (e.g., “image stabilization”), *named* (e.g., “SteadyShot feature”), *pronominal* (e.g., “it”), or *implicit* mentions (e.g., “reduction of blurring from camera shake”). We explicitly restrict the task to finding nominal aspect mentions[1].

The contribution of this paper is to explicitly cast the problem setting as a *terminology extraction* (TE) task and to examine the utility of methods that have been proven beneficial in this context. Most related work does not consider this close relationship and rather presents ad-hoc approaches. Our main contributions are as follows:
– We experiment with varying term acquisition methods, propose a set of new term filtering approaches, and consider variant aggregation techniques typically applied in TE systems.
– We compare the utility of different term relevance measures and experiment with combinations of these measures.
– We propose and assess a new method that filters erroneous modifiers (adjectives) in term candidates. Our method exploits information obtained from pros/cons summaries of customer reviews.
– Our best configuration improves over a state-of-the-art baseline by up to 7 percentage points.

The remainder of the paper is organized as follows: In Section 2, we cover related work, setting focus on unsupervised approaches. Section 3 describes the TE methods we examine in this study. Section 4 introduces our evaluation datasets and Section 5 presents experiments and results. We summarize and conclude in Section 6.

[1] Nominal mentions account for over 80% of all mentions in our datasets. Also in other corpora, the ratio is quite similar, e.g., (Kessler et al., 2010).

Proceedings of the Seventeenth Conference on Computational Natural Language Learning, pages 222–230,
Sofia, Bulgaria, August 8-9 2013. ©2013 Association for Computational Linguistics

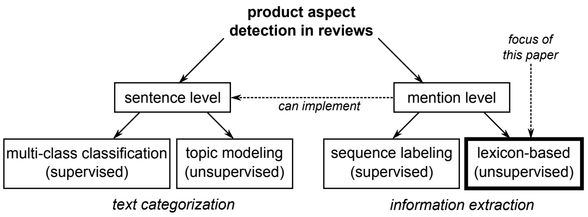

Figure 1: Conceptual overview of related work in product aspect detection.

2 Related Work

Figure 1 provides a conceptual overview of different tasks and approaches in the research area. Basically, we differentiate related work by the granularity of analysis, distinguishing between sentence level and mention level analysis. While at the sentence level, the goal is to decide whether a given sentence refers to one or more predefined aspects, fine-grained mention level analysis aims at discovering each individual mention of a relevant product aspect (e.g., “The *image stabilization* works well, but I didn’t like the poor *battery life*.”).

We address aspect detection at the **mention level** and our methods fall into the category of (unsupervised) **lexicon-based approaches**. In contrast to supervised methods, lexicon-based approaches do not rely on labeled training data and thus scale better across domains[2]. The common approach is to crawl a corpus of reviews and to apply frequency-based methods to extract a lexicon of product aspects from the dataset. Approaches differ in the way corpus statistics are computed and to which extent linguistic features are exploited. Section 2.1 briefly describes the most relevant previous works and Section 2.2 provides an assessment of the different approaches.

2.1 Creating Product Aspect Lexicons

Hu and Liu (2004) cast the problem as a **frequent itemset mining** task and apply the well-known *Apriori algorithm* (Agrawal and Srikant, 1994). Inherent drawbacks of this approach[3] are heuristically treated in a post-processing step.

Whereas Hu and Liu’s method exclusively examines documents of the input collection, Popescu and Etzioni (2005) propose to incorporate the **Web as a corpus**. They assess a term candidate’s domain relevance by computing the *pointwise mutual information* (PMI) (Zernik, 1991) between the candidate term and some predefined phrases that are associated with the product type. The *PMI* score is used to prune term candidates.

A further approach is to utilize a **contrastive background corpus** to determine the domain relevance of terms. For instance, Yi et al. (2003) use the *likelihood ratio test* (LRT) to compute a confidence value that a term candidate originates from the relevant review corpus. The computed score is used to rank term candidates. Also Scaffidi et al. (2007) follow the basic idea of using a contrastive corpus, but simply compare relative frequency ratios instead of computing a confidence value. Other exemplary works consider the utility of **statistical language models** (Wu et al., 2009), propose **latent semantic analysis** (Guo et al., 2009), or examine a **double propagation approach** that leverages the correlation between product aspects and sentiment bearing words (Zhang et al., 2010). Product aspect lexicons may also be created manually, e.g., Carenini et al. (2005) or Bloom et al. (2007) follow this approach. Naturally, a manual approach does not scale well across domains.

2.2 Assessment of Lexicon-Based Approaches

Our goal in this section is to select a state-of-the art method that we can use as a baseline in our experiments. Unfortunately, it is quite difficult to assess the relative performance of the different approaches as the evaluation datasets and methodologies often vary. Popescu and Etzioni (2005) compare their results to the method by Hu and Liu (2004) and report significantly improved results. However, their method relies on the private “Know-it-all” information extraction system and is therefore not suited as a baseline. Scaffidi et al. (2007) only assess the precision of the extracted aspect lexicon. Their methodology does not allow to measure recall, which renders their comparison to Hu’s method rather useless[4]. Furthermore, the results are quite questionable as the number of extracted aspects is extremely small (8-12 aspects compared to around thousand with our approach). Also Yi et al. (2003) only report results of an intrinsic evaluation for their LRT-approach. A systematic comparison of Hu’s frequent itemset min-

[2] For instance, (Jakob and Gurevych, 2010) report that F-scores for their sequence labeling method decrease by up to 25 percentage points in cross domain settings.

[3] The word order is not recognized and sub-terms of terms are not necessarily valid terms in natural language.

[4] Without considering recall, the precision can easily be tweaked by adjusting threshold values.

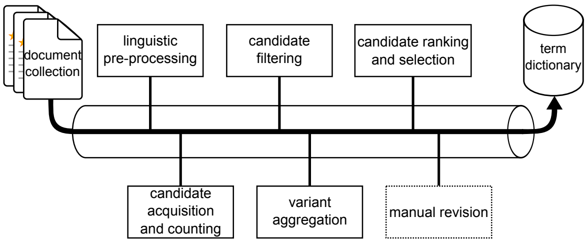
document collection
linguistic pre-processing
candidate filtering
candidate ranking and selection
term dictionary
candidate acquisition and counting
variant aggregation
manual revision

Figure 2: Pipeline architecture of a TE system.

ing and Yi's LRT-approach is conducted by Jakob (2011). His results show that "the Likelihood Ratio Test based approach generally yielded better results". In the absence of other valid comparative studies, we therefore select the LRT-approach as a baseline method for our experiments.

3 Terminology Extraction for Product Aspect Detection

A typical TE system follows the pipeline architecture depicted in Figure 2. Depending on the specific application domain, the implementation of the individual pipeline steps may differ widely. For example, we will see in the next section that the examined acquisition and filtering methods are highly tailored to the domain of customer reviews. In contrast, the underlying concepts for the definition of term relevance are applicable across domains. From the multitude of statistical measures proposed in the literature[5], we can distill mainly three underlying concepts: (1) *contrastive domain relevance*, (2) *intra domain relevance*, and (3) *term cohesion*. We will experiment with measures for all of the three concepts. The following subsections describe how we implement the individual steps of the extraction pipeline (for the majority of steps, we propose several alternative approaches, which will be subject to experimentation).

3.1 Linguistic Preprocessing

We preprocess all text documents by means of a *part-of-speech tagger*[6] (which also performs tokenization, sentence splitting, and lemmatization). All tokens are further normalized by case folding.

3.2 Candidate Acquisition

The candidate acquisition component initially decides which phrases are further considered and which are directly discarded. Defining too restrictive filters may lower the recall, whereas too unconstrained filters may decrease the precision.

Part-of-Speech Tag Filter We experiment with two POS-tag filters: *BNP1* and *BNP2*. As a baseline (BNP1), we use the "base noun phrase pattern" proposed in (Yi et al., 2003):

```
BNP1 := NN |NN NN |JJ NN |NN NN NN |
        JJ NN NN |JJ JJ NN
```

It restricts candidates to a maximum length of three words (adjectives or nouns), where adjectives must only occur as pre-modifiers to nouns. As an alternative, we examine the utility of a more relaxed pattern (BNP2). This pattern matches terms of arbitrary length, also allows for plural forms, and matches proper nouns (identified by the tags NNP or NNPS):

```
BNP2 := (JJ )*(NN\w{0,2} )+
```

Domain Specific Heuristics Acquisition heuristics put further constraints on the validity of term candidates. As a baseline, we consider two heuristics proposed in (Yi et al., 2003):

– The *definite base noun phrase* (**DBNP**) heuristic restricts the *BNPs* to phrases that are preceded by the definite article "*the*".

– The *beginning definite base noun phrase* (**BBNP**) heuristic restricts valid candidates to *DBNPs* that occur at the beginning of a sentence, followed by a verb phrase (e.g., "The *picture quality* is great.").

As an alternative, we propose two other heuristics. Both are based on the hypothesis that the occurrence of sentiment expressions in the context of a candidate is a good indicator for the candidate's validity. Sentiment expressions are detected with a small hand-crafted sentiment lexicon composed of 520 strongly positive/negative adjectives. We experiment with two different strategies:

– The *sentiment bearing sentence* (**SBS**) heuristic only considers candidates that occur in sentences where at least one sentiment expression is detected.

– The *sentiment bearing pattern* (**SBP**) heuristic defines a set of four simple syntactic patterns that relate candidate terms to sentiment expressions. Only candidates that match one of the patterns are further considered.

3.3 Candidate Filtering

Although the candidate acquisition heuristics focus on high precision, they generate a consider-

[5] For example, consult (Kageura and Umino, 1996) for a thorough literature survey on terminology extraction.

[6] http://nlp.stanford.edu/software/corenlp.shtml

able number of irrelevant candidates. These can be pruned by further domain specific filters:

Review Stop Word Filter We compile a list of review specific *stop words* and discard each candidate term that contains at least one of the words. The list (176 entries) has been constructed based on observations on a development dataset and by (intelligent) extrapolation of these findings. Roughly categorized, it includes *sentiment bearing nouns* (e.g., “complaint”), *review related terms* (e.g., “bottom line”), *purchase related phrases* (e.g., “delivery”), *mentions of persons* (e.g., “wife”), and *phrases of reasoning* (e.g., “decision”).

Pre-Modifier Filter Both presented part-of-speech filters (BNP1/2) allow nouns to be modified by multiple adjectives. Unfortunately, this leads to the extraction of many invalid terms (e.g., “great/JJ design/NN” or “new/JJ design/NN”). Quite frequently, *sentiment bearing adjectives* such as “great”, “fantastic”, or “bad” are erroneously extracted. We utilize our hand-crafted sentiment lexicon to prune these modifiers. Another type is related to adjectives that act as *universal modifiers* in terms (e.g., “new”, “long”, or “red”). For such adjectives we cannot compile a stop word list. We experiment with two different methods for filtering universal modifiers. As a baseline, we examine a filter proposed by Kozakov et al. (2004) as part of their *GlossEx* glossary extraction system. As a second approach, we propose a method that uses signals from pros/cons summaries of reviews (Section 3.6).

Product Name Filter As we are only interested in finding nominal aspect mentions, we need to discard all candidate terms that refer to product or brand names. For this purpose, we automatically generate a stop word list by exploiting meta data (on products and brands) that is associated with the crawled customer reviews. Whenever a term candidate contains a token that is present in the appropriate stop word list, the candidate is discarded.

3.4 Variant Aggregation

The goal of this step is to find all variants of a term and to identify a canonical representation. For example, the variants “auto-focus”, “auto focus”, “autofocus”, or “auto focuss” should be mapped to the canonical form “auto focus”. The purpose of this step is twofold: (1) higher lexicon coverage and (2) preventing potential problems with data sparseness during candidate ranking. Following Kozakov et al. (2004), we implement heuristics for finding *symbolic*, *compounding*, and *misspelling* variants. In addition, we implement a method that considers *compositional* variants of the form “room size” vs. “size of the room”.

3.5 Candidate Ranking and Selection

Candidate ranking is at the core of each terminology extraction system. As it is unclear which relevance measure performs best in our context, we experiment with different approaches and also consider reasonable combinations of individual scores. Despite the newly proposed *diversity value* score, the selected measures are all taken from previous research in terminology extraction. We therefore only briefly discuss the other measures and refer to the original literature for more details.

Raw Frequency (Intra Domain) The ranking is simply determined by the raw occurrence frequency of a term.

Relative Frequency Ratio (Contrastive) This ranking (MRFR) is based on the comparison of relative frequency ratios in two corpora. While the original measure (Damerau, 1993) is only defined for single word terms, Kozakov et al. (2004) show how to extend the definition to multi-word terms.

Likelihood Ratio Test (Contrastive) This ranking can be considered as a more robust version of the MRFR approach. Put simply, it additionally computes confidence scores for the relative frequency ratios, which allows to prevent problems with low frequency terms. The score is based on the *likelihood ratio test* (LRT). Yi et al. (2003) describe how the score is computed in our context.

Generalized Dice Coefficient (Term Cohesion) To measure the association between words of a complex term, Park et al. (2002) introduce a measure that generalizes the *Dice coefficient* (Dice, 1945). The measure gives higher scores to terms with high co-occurrence frequencies.

Diversity Value (Intra Domain) Based on the observation that nested word sequences that appear frequently in longer terms are likely to represent the key parts or features of a product, we propose a measure that gives higher scores to such “key terms” (e.g., “lens” occurs in terms such as “autofocus lens”, “zoom lens”, “macro lens”,

"lens cap", or "lens cover"). Inspired by the *C-Value score* (Frantzi and Ananiadou, 1996), we define the measure as: *diversity-score(ws)* =

$$\log_2(|ws|_t + 1) * \frac{\sum_{w_i \in ws} (f(w_i) * \log_2(|T^*_{w_i}| + 1))}{|ws|_t},$$

where $|ws|_t$ denotes the number of tokens of a word sequence ws, w_i refers to the i-th token in ws, and $T^*_{w_i}$ describes the set of other candidate terms that contain the token w_i. The function $f(w_i)$ returns the frequency of the token w_i in the considered text corpus.

Combining Ranking Measures

As the presented ranking measures are based on different definitions of term significance, it is reasonable to compute a combined score (e.g., combining a term's contrastive relevance with its strength of cohesion). Since the different measures are not directly comparable, we compute a combined score by considering the individual rankings: Let T be the set of extracted candidate terms and let $R_i(t)$ be a function that ranks candidates $t \in T$. Using a weight ω_i for each of the n selected measures, we compute the final rank of a candidate t as: *weighted-rank*(t) =

$$\sum_{i=1}^{n} \omega_i * R_i(t)\ \text{, where } \sum_{i=1}^{n} \omega_i = 1.$$

For our experiments, we chose equal weights for each ranking measure, i.e., $\omega_i = 1/n$.

3.6 Pros/Cons Pre-Modifier Filter

Some sentiment bearing pre-modifiers are domain or aspect-specific (e.g., "*long* battery life")[7]. The *GlossEx filter* (see Section 3.3) cannot cope with this type of modification. To identify such pre-modifiers, we propose to exploit signals from *structured* pros/cons summaries that typically accompany a customer review. We hypothesize that valid pre-modifiers (e.g., "digital" in "digital camera") occur similarly distributed with their head noun in both, lists of pros and lists of cons. Invalid pre-modifiers, i.e., aspect-specific sentiment words, are likely to occur either more often in lists of pros or lists of cons. We design a simple *likelihood ratio test* to operationalize this assumption.

In particular, we consider the probabilities $p_1 = Pr(pm|head;\textit{pros})$ and $p_2 = Pr(pm|head;\textit{cons})$, where p_1 (p_2) denotes the probability in a corpus of pros (cons) lists that pm occurs as pre-modifier with the head noun $head$.

statistic	hotel	camera
documents	150	150
sentences	1,682	1,416
tokens	29,249	24,765
nominal aspect mentions (incl. sentiment targets)	2,066	1,918
avg. tokens per mention	1.28	1.4
distinct mentions	490	477

Table 1: Basic corpus statistics.

To design a hypothesis test, we assume as null hypothesis H_0 that $p_1 = p = p_2$ (equal distribution in pros and cons) and as alternative hypothesis that $p_1 \neq p_2$ (unequal distribution). We calculate the likelihood ratio λ and utilize the value $-2 * log\lambda$ to reject H_0 at a desired confidence level (in that case, we prune the pre-modifier pm).

4 Datasets

We evaluate our approaches on datasets of hotel and digital camera reviews. We crawled around 500,000 hotel reviews from Tripadvisor.com and approximately 200,000 digital camera reviews from Amazon.com, Buzzillions.com, and Epinions.com. From each of the two crawls, we randomly sample 20,000 reviews, which we use as foreground corpora for the terminology extraction task[8]. As a background corpus, we utilize a 100,000 document subset (randomly sampled) of the "ukWaC corpus" (Baroni et al., 2009).

4.1 Evaluation Corpora

To evaluate our approaches, we manually annotate a subset of the crawled reviews. In particular, we randomly sample subsets of 150 hotel and 150 camera reviews that do not overlap with the foreground corpora. Following prior work on sentiment analysis (Wiebe et al., 2005; Polanyi and Zaenen, 2006), we decompose an opinion into two functional constituents: *sentiment expressions* and *sentiment targets*. In addition, we consider *nominal mentions* of product aspects that are not targeted by a sentiment expression. We annotate a document by marking relevant spans of text with the appropriate annotation type, setting the type's properties (e.g., the polarity of a sentiment expression), and relating the annotations to each other. Table 1 summarizes the statistics of the created evaluation corpora (regarding sentiment targets and nominal aspect mentions).

[7] see also (Fahrni and Klenner, 2008)

[8] Larger corpora did not improve our results.

5 Experiments and Results

5.1 Evaluation Methods

We conduct *intrinsic* and *extrinsic* evaluation of the approaches. Intrinsic evaluation refers to assessing the quality of the generated product aspect lexicons. For this purpose, we manually inspect the extracted lexicons and report results in terms of *precision* (share of correct entries) or *precision@n* (the precision of the *n* highest ranked lexicon entries). For extrinsic evaluation (evaluation in use), we apply the extracted lexicons for the task of aspect detection in customer review documents. To match lexicon entries in review texts, we apply the Aho-Corasick algorithm (Aho and Corasick, 1975). If multiple matches overlap, we select the left-most, longest-matching, highest-scoring lexicon entry (thus guaranteeing a set of non-overlapping matches). Only exact matches are counted as true positives. We further differentiate between two evaluation scenarios:

– Scenario A: In this scenario, the task is to extract all product aspects, irrespective of being target of a sentiment expression or not. We thus define the union of sentiment target and aspect mention annotations as reference (gold standard). Any extraction that matches either a sentiment target or an aspect mention is considered a true positive.

– Scenario B: This scenario considers the task of detecting sentiment targets. As it is not our goal to assess the accuracy of sentiment expression detection, we provide the extraction algorithm with perfect (gold standard) knowledge on the presence of sentiment expressions and their relations to sentiment targets (in effect, the algorithm only considers matches that overlap a sentiment target).

5.2 Baseline Results (Yi et al. Method)

To make our results comparable to other existing methods, we first set a baseline by applying a state-of-the-art approach on our datasets. As motivated in Section 2.2, the LRT-approach by Yi et al. (2003) represents our baseline. We can easily implement Yi's method with our terminology extraction framework by using the *BNP1* POS-tag filter, the *bBNP* acquisition heuristic, and the LRT-score for ranking. We select all terms with a minimum LRT-score of 3.84[9] and do not apply any candidate filtering or variant aggregation.

scenario	precision	recall	f-measure
hotel A	55.1%	73.0%	62.8%
hotel B	81.3%	71.2%	75.9%
camera A	65.0%	72.5%	68.6%
camera B	76.8%	69.9%	73.2%

Table 2: Extrinsic evaluation results for the baseline approach.

scenario	precision	recall	f-measure
hotel A	56.9% (+1.8*)	75.2% (+2.2*)	64.8% (+2.0*)
hotel B	85.7% (+4.4*)	75.1% (+3.9*)	80.0% (+4.1*)
camera A	69.2% (+4.2*)	74.3% (+1.8*)	71.7% (+3.1*)
camera B	79.3% (+2.5*)	72.2% (+2.3*)	75.6% (+2.4*)

Table 3: Results with activated candidate filters.

The baseline method produces lexicons with 1,182 (hotel) and 953 (digital camera) entries. Due to our significantly larger foreground corpora, the dictionaries' sizes are by far larger than reported by (Yi et al., 2003) or by (Ferreira et al., 2008). Intrinsic evaluation of the lexicons reveals precision values of 61.2% (hotel) and 67.6% (camera). For precision@40, we find values of 62.5 (hotel) 80.0 (camera).

Table 2 reports the extrinsic evaluation results for the baseline configuration. Naturally, the precision values obtained for scenario A are lower than for the "synthetic" scenario B (where partial matches are the only possible source for false positives). Recall values in both scenarios are moderately high with around 70%.

If not otherwise stated, the configurations in the following sections apply the BNP1 acquisition pattern, the *BBNP* heuristic, and the LRT-ranking with a minimum score of 3.84.

5.3 Effectiveness of Candidate Filtering

In this section, we analyze the influence of candidate filtering (baseline: Yi's method). When applying all filters jointly (except for the pros/cons filter), the resulting lexicons consist of 975 (hotel) and 767 (camera) entries. Compared to the baseline, the (intrinsic) precision of the lexicons improves by around 10 percentage points (hotel) and 14 percentage points (camera). Each individual filter has a positive effect on the precision, where the *GlossEx filter* has the greatest influence (+5 percentage points in both corpora). Table 3 shows that the improved lexicon precision also leads to better results for the product aspect extraction task. The observed f-measure values increase by up to 4.1 percentage points compared to the baseline

[9] 3.84 is the critical value of the χ^2-distribution for one degree of freedom at a confidence level of 95%.

scenario	precision	recall	f-measure
hotel A	56.7% (-0.2)	75.1% (-0.1)	64.6% (-0.2)
hotel B	85.5% (-0.2)	75.1% (0.0)	79.9% (-0.1)
camera A	69.8% (+0.6)	74.8% (+0.5)	72.2% (+0.5)
camera B	80.7% (+1.4)	73.0% (+0.8)	76.7% (+1.1)

Table 4: Results with variant aggregation.

method. All improvements are statistically significant[10]. The increase in recall is mainly due to successful pruning of false modifiers.

5.4 Effectiveness of Variant Aggregation

In this section, we examine the influence of the different variant aggregation techniques (baseline: Yi's method + filter). To assess the effectiveness of variant aggregation, we only evaluate extrinsically (since we primarily expect a higher coverage of the lexicons). Table 4 compares the results with variant aggregation to the results of the previous section (all filters activated). The results show that variant aggregation has only marginal effects. Although we can measure improved results for the camera corpus, the differences are rather small and not statistically significant. For the hotel corpus, the influence is even lower. To understand the reasons for the insignificant effect, we perform a mistake analysis of the false negatives in scenario B. In particular, we compare the false negatives with and without variant aggregation. For the hotel corpus, we only find 18 out of 251 false negatives (7.2%) that are candidates for variant aggregation. In the ideal case (variant aggregation successfully recognizes all the candidates), this translates to a maximum gain of 1.8 percentage points in recall. For the camera dataset, we calculate a maximum gain of 2.4 percentage points. Our results deviate from the ideal case for mainly two reasons: (1) Most variants occur rarely and the ones that occur in the evaluation corpora do not occur in the foreground corpora. (2) Some variants (e.g., misspellings) are so frequent in the foreground corpus that the LRT-ranking already selects them as independent terms.

5.5 Influence of Acquisition Methods

This section examines the influence of the different acquisition patterns and heuristics. We only report results for the hotel dataset as the results for the camera corpus are similar. Table 5 shows results for scenario B (all filters and aggregation methods activated). As could be expected, the more relaxed acquisition pattern *BNP2* trades precision for an increased recall (+1-2 percentage points). The results further show that the use of appropriate acquisition heuristics is quite important. We can improve the f-measure by up to 6.1 percentage points. We find that the *SBP* and *BBNP* heuristics perform best on our datasets. The differences in f-measure, compared to the other two heuristics, are statistically significant (not shown in the table). As the *BBNP* heuristic is easier to implement and shows comparable results, we conclude that it is preferable over the *SBP* method.

	precision		recall		f-measure	
heuristic	BNP1	BNP2	BNP1	BNP2	BNP1	BNP2
—	80.7%	79.5%	70.7%	71.7%	75.4%	75.4%
SBS	81.1%	80.0%	72.2%	72.9%	76.4%	76.3%
DBNP	83.2%	82.4%	73.6%	75.2%	78.1%	78.6%
SBP	**87.0%**	84.5%	74.6%	75.8%	**80.3%**	79.9%
BBNP	85.5%	**85.5%**	**75.1%**	**77.7%**	79.9%	**81.5%**

Table 5: Extrinsic evaluation results with varying acquisition patterns and heuristics (hotel dataset).

	hotel		camera	
measure	precision	p@40	precision	p@40
frequency	41.6%	55.0%	44.8%	70.0%
dice	39.0%	55.0%	43.5%	87.5%
diversity	66.4%	77.5%	76.7%	70.0%
lrt	69.6%	72.5%	81.1%	87.5%
mrfr	72.0%	87.5%	81.4%	92.5%

Table 6: Intrinsic evaluation results with the five different ranking measures.

5.6 Influence of Ranking Functions

We now examine the influence of the different ranking measures (all filters and variant aggregation are activated). To rule out the influence of varying lexicon sizes, we choose a fixed size for each dataset (determined by the number of terms that exhibit an LRT-score greater than 3.84). For larger lexicons, we prune the entries with the lowest scores. For each configuration, we apply all filter and variant aggregation approaches. Table 6 shows the intrinsic evaluation results. We can clearly observe that the contrastive relevance measures (LRT and MRFR) outperform the intra domain and term cohesion measures. The MRFR-ranking shows better results than the LRT-ranking in both corpora, especially w.r.t. precision@40.

The improved results with contrastive measures are also reflected by our extrinsic evaluation. Ta-

[10] We use the * notation to indicate statistically significant differences. If not otherwise stated, significance is reported at the 99% confidence level.

	hotel			camera		
measure	prec.	rec.	F	prec.	rec.	F
frequency	45.3%	79.1%	57.6%	50.7%	77.8%	61.4%
dice	44.7%	78.3%	56.9%	50.4%	77.5%	61.1%
diversity	51.4%	72.3%	60.1%	64.5%	73.8%	68.8%
lrt	56.7%	75.1%	**64.6%**	69.8%	74.8%	72.2%
mrfr	**60.2%**	67.3%	63.5%	**73.1%**	72.8%	**73.0%**
all	46.6%	**79.3%**	58.7%	52.6%	**78.7%**	63.0%
mrfr-dice	47.7%	78.2%	59.2%	55.3%	78.2%	64.8%
lrt-div.	47.8%	73.5%	57.9%	57.0%	75.1%	64.8%
mrfr-lrt	56.9%	73.5%	64.2%	68.2%	73.7%	70.8%
mrfr-freq.	51.8%	77.8%	62.2%	61.2%	76.1%	67.9%
mrfr-lrt-div.	53.3%	74.5%	62.2%	66.8%	75.4%	70.8%
mrfr-div.	57.9%	73.1%	**64.6%**	71.8%	72.5%	72.1%

Table 7: Extrinsic evaluation results for varying ranking methods (scenario A).

scenario	precision	recall	f-measure
hotel A	58.0% (+1.1*)	76.3% (+1.1)	65.9% (+1.1*)
hotel B	88.9% (+3.2*)	77.4% (+2.4*)	82.8% (+2.8*)
camera A	71.7% (+2.5*)	76.2% (+1.9)	73.9% (+2.2*)
camera B	83.4% (+4.0*)	75.1% (+2.9*)	79.0% (+3.4*)

Table 8: Results with active pros/cons filter.

ble 7 presents the results for scenario A, considering the measures in isolation and in selected combinations (using equal weights). Compared to raw frequency, the contrastive measures exhibit f-measure values that are between 7 (hotel) and 11.6 (camera) percentage points higher. We hypothesized that the combination of different relevance concepts (e.g., contrastive + term cohesion) could improve the system's performance, but the obtained results do *not* confirm this hypothesis.

5.7 Effectiveness of Pros/Cons Filter

In this section we examine the the pros/cons premodifier filter. The reported results are based on pros/cons corpora composed of 100,000 (hotel) and 50,000 (camera) documents. We set the threshold for the hypothesis test to 10.83, corresponding to a 99.9% confidence level. Table 8 presents the results of additionally applying this filter (baseline: all other filters and variant aggregation activated). We can observe statistically significant improvements with gains in f-measure of up to 3.4 percentage points. Examining the resulting lexicons, we find that the filter successfully pruned around 40 false pre-modifiers, which increases the (intrinsic) precision by around 3 percentage points for both datasets. Despite the relatively few lexicon entries that are altered by means of the filter, we observe the mentioned (significant) gains in f-measure. For both datasets this is mainly because the affected lexicon entries exhibit a high occurrence frequency in the evaluation datasets (e.g., "large room" or "low price").

6 Conclusions

Identifying the most relevant aspects of a given product or product type constitutes an important subtask of an aspect-based review mining system. In this work, we explicitly cast the task as a terminology extraction problem. We were interested whether methods that have been proven beneficial in TE systems also help in our application scenario. Additionally, we proposed and evaluated some new term acquisition heuristics, candidate filtering techniques, and a ranking measure. The results show that our terminology extraction approach allows to generate quite accurate product aspect lexicons (precision up to 85%), which in turn allow for f-measures of up to 74% for an aspect detection task and up to 83% for a (synthetic) sentiment target detection task. Compared to a relevant baseline approach (Yi et al., 2003), we observe increases in f-measure by 3-7 percentage points for different evaluation scenarios.

With regard to the different configurations of our system, we made the following observations:
– Improved results are mainly due to the proposed candidate filtering techniques. Each individual filter has been found to be beneficial. The proposed pros/cons filter raised the f-measure by up to 3.4 percentage points.
– The choice of the acquisition heuristic is important. We measured differences of up to 6.1 percentage points in f-measure. The *SBP* and *BBNP* heuristics performed best. The relaxed *BNP2* pattern increases the recall and is a reasonable choice if extracted lexicons are manually post-processed.
– The variant aggregation techniques had only a marginal effect.
– The contrastive relevance measures *LRT* and *MRFR* performed best. Neither the proposed *diversity value* score, nor combinations of different relevance measures proved to be beneficial.
– In summary, we suggest to use the *BNP2* acquisition pattern and the *BBNP* or *SBP* acquisition heuristic, to activate all mentioned filters, and to use a contrastive relevance measure for ranking. Whereas variant aggregation was not beneficial within the TE pipeline, it is nonetheless important and should be considered downstream, i.e., during application of the extracted lexicons.

References

Rakesh Agrawal and Ramakrishnan Srikant. 1994. Fast algorithms for mining association rules in large databases. In *Proceedings of the 20th VLDB*, pages 487–499, San Francisco, CA, USA.

Alfred V. Aho and Margaret J. Corasick. 1975. Efficient string matching: An aid to bibliographic search. *Comm. of the ACM*, 18(6):333–340.

M. Baroni, S. Bernardini, A. Ferraresi, and E. Zanchetta. 2009. The WaCky Wide Web: A collection of very large linguistically processed web-crawled corpora. *LREC*, 43:209–226.

K. Bloom, N. Garg, and S. Argamon. 2007. Extracting appraisal expressions. In *Proceedings of the NAACL HLT 2007*, pages 308–315. ACL.

G. Carenini, R. T. Ng, and E. Zwart. 2005. Extracting knowledge from evaluative text. In *Proceedings of the K-CAP '05*, pages 11–18. ACM.

F. J. Damerau. 1993. Generating and evaluating domain-oriented multi-word terms from texts. *Information Proc. and Management*, 29(4):433–447.

Kushal Dave, Steve Lawrence, and David M. Pennock. 2003. Mining the peanut gallery: Opinion extraction and semantic classification of product reviews. In *Proceedings of the WWW '03*, pages 519–528, New York, NY, USA. ACM.

Lee R. Dice. 1945. Measures of the amount of ecologic association between species. *Ecology*, 26(3).

A. Fahrni and M. Klenner. 2008. Old wine or warm beer: Target-specific sentiment analysis of adjectives. In *Symposion on Affective Language in Human and Machine, AISB Convention*, pages 60–63.

L. Ferreira, N. Jakob, and I. Gurevych. 2008. A comparative study of feature extraction algorithms in customer reviews. In *Proceedings of the 2008 International Conference on Semantic Computing*, pages 144–151. IEEE Computer Society.

K. T. Frantzi and S. Ananiadou. 1996. Extracting nested collocations. In *Proceedings of the 16th COLING*, pages 41–46. ACL.

H. Guo, H. Zhu, Z. Guo, X. Zhang, and Z. Su. 2009. Product feature categorization with multilevel latent semantic association. In *Proceedings of the 18th CIKM*, pages 1087–1096. ACM.

Minqing Hu and Bing Liu. 2004. Mining and summarizing customer reviews. In *Proceedings of the 10th ACM SIGKDD*, pages 168–177. ACM.

N. Jakob and I. Gurevych. 2010. Extracting opinion targets in a single- and cross-domain setting with conditional random fields. In *Proceedings of the EMNLP '10*, pages 1035–1045. ACL.

Niklas Jakob. 2011. *Extracting Opinion Targets from User-Generated Discourse with an Application to Recommendation Systems*. Ph.D. thesis, Technische Universtität Darmstadt.

K. Kageura and B. Umino. 1996. Methods of automatic term recognition: A review. *Terminology*, 3(2):259–289.

J. S. Kessler, M. Eckert, L. Clark, and N. Nicolov. 2010. The 2010 ICWSM JDPA sentiment corpus for the automotive domain. In *Proceedings of the 4th AAAI Conference on Weblogs and Social Media Data Workshop Challenge*.

L. Kozakov, Y. Park, T. Fin, Y. Drissi, Y. Doganata, and T. Cofino. 2004. Glossary extraction and utilization in the information search and delivery system for IBM technical support. *IBM Systems Journal*, 43(3):546–563.

R. Navigli and P. Velardi. 2004. Learning domain ontologies from document warehouses and dedicated web sites. *Comp. Linguistics*, 30(2):151–179.

Youngja Park, Roy J. Byrd, and Branimir K. Boguraev. 2002. Automatic glossary extraction: Beyond terminology identification. In *Proceedings of the 19th COLING*, pages 1–7. ACL.

Livia Polanyi and Annie Zaenen. 2006. Contextual valence shifters. In *Computing Attitude and Affect in Text: Theory and Applications*, volume 20 of *The Information Retrieval Series*, chapter 1, pages 1–10. Springer Netherlands, Berlin/Heidelberg.

A.-M. Popescu and O. Etzioni. 2005. Extracting product features and opinions from reviews. In *Proceedings of HLT EMNLP '05*, pages 339–346. ACL.

C. Scaffidi, K. Bierhoff, E. Chang, M. Felker, H. Ng, and C. Jin. 2007. Red opal: product-feature scoring from reviews. *Proceedings of the 8th ACM Conference on Electronic Commerce*, pages 182–191.

J. Wiebe, T. Wilson, and C. Cardie. 2005. Annotating expressions of opinions and emotions in language. *LREC*, 39(2):165–210, May.

Y. Wu, Q. Zhang, X. Huang, and L. Wu. 2009. Phrase dependency parsing for opinion mining. In *Proceedings of the EMNLP '09*, pages 1533–1541. ACL.

J. Yi, T. Nasukawa, R. Bunescu, and W. Niblack. 2003. Sentiment Analyzer: Extracting sentiments about a given topic using natural language processing techniques. In *Proceedings of the 3rd ICDM*, pages 427–434. IEEE Comput. Soc.

U. Zernik. 1991. *Lexical Acquisition: Exploiting Online Resources to Build a Lexicon*. Lawrence Erlbaum.

L. Zhang, B. Liu, S. H. Lim, and E. O'Brien-Strain. 2010. Extracting and ranking product features in opinion documents. In *Proceedings of the 23rd COLING: Posters*, pages 1462–1470. ACL.