Revised Mutual Information Approach for German Text Sentiment Classificatio

Farag Saad
GESIS - Leibniz Institute for the Social Sciences
Unter Sachsenhausen 6-8
50667 Cologne, Germany
farag.saad@gesis.org

Brigitte Mathiak
GESIS - Leibniz Institute for the Social Sciences
Unter Sachsenhausen 6-8
50667 Cologne, Germany
brigitte.mathiak@gesis.org

ABSTRACT

The significant increase in content of online social media such as product reviews, blogs, forums etc., have led to an increasing attention to sentiment analysis tools and approaches that make use of mining this substantially growing content. The aim of this paper is to develop a robust classification approach of customer reviews based on a self-annotated domain-specific corpus by applying a statistical approach i.e., mutual information. First, subjective words in each test sentence are identified. Second, ambiguous adjectives such as high, low, large, many etc., are disambiguated based on their accompanying noun using a conditional mutual information approach. Third, a mutual information approach is applied to find the sentiment orientation (polarity) of the identified subjective words based on analyzing their statistical relationship with the manually annotated sentiment labels within a sizeable sentiment training data. Fourth, since negation plays a significant role in flipping the sentiment polarity of an identified sentiment word, we estimate the role of negation in affecting the classification accuracy. Finally, the identified polarity for each test sentence is evaluated against experts' annotation.

Categories and Subject Descriptors

H.3.3 [**Information Storage and Retrieval**]: Information Search and Retrieval, Selection process, Information filtering;

General Terms

Algorithms, Experimentation

Keywords

sentiment analysis, mutual information, negation, disambiguation

1. INTRODUCTION

Due to the continuing increase in internet content created for different purposes e.g., e-commerce, newswire websites, social websites, etc., the users' interest in the generated content, within these websites, continues to grow significantly. For example, e-commerce online shops have been growing dramatically in recent years, along with the growing number of customers reviewing products and services. This leads in turn, to many people relying, to a large extent, on the reviews of other customers who already use this product before they themselves buy any product or use any services. However, to a large extent, users generated contents in most websites, are left without organization and are stored in an improper structure. Furthermore, many products can have many reviews, with some being rather long, making it almost impossible for a customer to mine all reviews that can support his/her purchases decision. Therefore, mining these reviews automatically, in order to efficiently find each opinionated sentence is an important and difficult task to achieve. In this regard, sentiment analysis (sometimes called opinion mining) automatically analyzes customers' opinions on a specific product and finds out whether a review is positive, negative or neutral [22]. The sentiment might be expressed explicitly or implicitly depending on the formulation of the customers' opinions. In explicit sentiment, a piece of text contains a subjective sentence/sentences that express a clear opinion e.g., "poor picture quality" while an implicit sentiment is a piece of text which implies an opinion even though there are no sentiment bearing words within it, e.g., "The camera battery lasted for 1 hours".

Sentiment Analysis is an interdisciplinary task that spans across a number of different disciplines such as natural language processing and text mining. The intersection between these disciplines has led to the emergence of different challenges that need to be addressed for effective sentiment analysis. In order to alleviate these issues, sentiment analysis incorporates different subtasks e.g., subjectivity detection, polarity detection and sentiment strength detection [18]. In subjectivity detection, a customer's review is firstly segmented into several sentences, in order to find out which sentences are bearing sentiment e.g., the sentence, "The video capability is truly amazing" bearing positive sentiment. Furthermore, any objective sentences will be excluded from the sentiment analysis process. For example, the objective sentence "Yesterday I bought a Nikon camera" will be excluded as it expresses fact and not sentiment. After classifying a piece of text whether it is subjective or objective, the next step is the polarity detection. In polarity detection a given piece of text is classified either positive or negative based on the sentiment orientation of the words it contains. However, there is a piece of text that can fall between positive or negative classes i.e., a review of a product can carry positive as well as negative opinions at the same time e.g., "With the exception of burst shooting, this camera's performance

WWW 2013 Companion, May 13–17, 2013, Rio de Janeiro, Brazil.
ACM 978-1-4503-2038-2/13/05.

is excellent". In order to distinguish between positive and negative, sentiment strength detection can be used in that additional sentiment classes can be utilized such as "strong positive", "weak positive", "weak negative", etc. Sentiment analysis allows for further detailed analysis rather than only identifying the reviews polarity against a particular product as a whole i.e., identifying the sentiment polarity based on the features (attributes) level instead of the object level. For example, instead of identifying a consumer opinion about a digital camera as a whole, a particular opinion about each feature of a digital camera such as picture quality, weight, lens etc., can be identified. This can give wide details about which advantages and disadvantages a particular product may have [19].

The aim of this paper is to develop a robust classification approach of customer reviews based on a self-annotated domain-specific corpus by applying a statistical approach. This is done in two main steps: First, subjective words in each test sentence are identified. Second, a mutual information approach is used to find the sentiment orientation (polarity) of identified subjective words based on the training annotated sentences within sizeable training data. Finally, the identified polarity for each test sentence is evaluated against an experts' annotation.

The remainder of this paper is structured as follows. In section 2, discussion of related work on sentiment analysis is presented. In section 3, our approach for detecting online review polarity is discussed in detail. The evaluation of the proposed approach is discussed in section 4. Finally, the conclusion and hints of future work are presented in section 5.

2. RELATED WORK

Sentiment analysis has been studied employing different levels of analysis, such as the word level e.g., [26], and the attribute level e.g., [21], the concept level e.g., [5] as well as the sentence and clause level e.g., [32] and the document level e.g., [25].

Typically, sentiment analysis approaches can be classified into two main approaches, the lexical-based approach (usually called dictionary-based approach) and the machine learning approach. The Lexical-based approach represents the approach that makes the use of prior sentiment that is employed in predefined sentiment dictionaries. Most of the initial work in sentiment analysis has focused on using the lexical-based approach (e.g., [29]) to classify a given sentence or phrase by inferring the sentiment class of each individual word. However, in the last few years, there is a growing interest in using the machine learning approach, which uses different classifiers trained on manually annotated data [10]. Each approach suffers from certain limitations. For example, the lexical-based approach is domain dependent, in that it is not feasible to use the same dictionary for different domains e.g., the word "silent" would be considered as positive when dealing with "washing machine" product reviews while it would be considered negative when dealing with product reviews of "audio speakers". On the other hand, the machine learning approach requires a significant human effort to annotate a substantial number of training examples used to train the classifiers.

In the lexical-based approach, the sentiment dictionary includes a list of subjective words with their prior sentiment polarity. Identifying the sentiment of a given piece of text, using the lexical-based approach, is quite simple. Through the use of the dictionary look-up method, the process starts by matching words in the test sentence against the sentiment dictionary entries. If a given word in the test sentence is found in the dictionary, its polarity is obtained. After iterating through all words in the test sentence and obtaining their prior polarity, the greater overall polarity either positive or negative will reflect the given sentence's opinion i.e., more opinion positive words means a positive review and negative words implies a negative review.

Sentiment dictionaries can be either created manually e.g., [29] or automatically using positive and negative seed words to expand the dictionary e.g., [30]. Tong (2001) proposed a system for detecting sentiment over time for online discussions about movies [29]. The given online movie discussions are classified either as positive or negative based on the sentiment phrases they contain. The system performs the sentiment detection firstly, by tracking the online movie discussions in order to find out which phrases bear sentiments. This is achieved by using a hand-built lexicon of phrases associated with sentiment labels.

A manually built lexicon has been used effectively by different researchers. For example, Hatzivassiloglou and Wiebe (2000) focused on using adjectives as a clue to the sentiment orientation of a given text [13]. Based on the manually created lexicon for adjectives and their semantic orientation values (SO), for any given text, all adjectives are extracted and associated with their dictionary SO values. The overall sentiment score is obtained by summing up all adjective SO scores within the given text. The given text is then classified as bearing a positive or negative sentiment based on the overall score for the obtained adjectives within it.

Using a hand-built dictionary would be significant in detecting a sentiment for a given domain. However, new domains necessitate the creation of new hand-built lexicons which is very labor intensive. In order, to create a sentiment dictionary efficiently and alleviate any manual effort, Turney (2002) proposed a simple promising approach to create a sentiment dictionary in an automatic way [30]. The construction of the dictionary is accomplished based on the utilization of seed words that belong either to a positive or negative sentiment class. In order to find the correlation between a seed word and the target word that will be included in the dictionary, a mutual information approach, based on statistical data extracted from the web using the AltaVista search engine, is used. The target word is submitted as a query to the search engine i.e., either with the word "excellent" or the word "poor". The semantic orientation is then obtained based on the mutual information between the target word with the word "excellent" or with the word "poor". If the obtained mutual information score for the target word with the word "excellent" is greater than the one with the word "poor", the target word will be classified as positive, otherwise it will be classified as negative.

Hu and Liu (2004) claimed that the created dictionary list can be further expanded by considering synonym and antonym sets in WordNet [23], presuming that the semantic similarity employs sentimental similarity [14]. The semantic orientation of groups of synonyms assumed to be similar e.g., "beautiful" and "pretty" while the semantic orientation of antonyms are supposed to be opposite e.g., "excited" and "bored". However, Leung et al. (2006), based on statistical evidence obtained from movie review data, argue that

semantic similarity does not necessarily employ sentimental similarity [17].

In conjunction with the creation of sentiment resources from scratch, publicly available resources such as WordNet-Affect [31], SenticNet [4] and SentiWordNet [12] can be used to extract the semantic and affective information associated with natural language concepts. For example, Esuli and Sebastiani (2006) proposed a SentiWordNet, a publicly available lexical resource for opinion mining, by adding polarity labels for each term which exists in the WordNet [12].

With the availability of the sentiment annotated data, machine learning techniques have been used for sentiment classification tasks, for example, Support Vector Machines (SVMs) ([25] and [2]), Naïve Bayes (NB) ([36] and [22]) and Maximum Entropy (ME) ([24] and [25]).

For advancing sentiment resource creation for the German language, Remus et al. (2010) studied the possibility of creating a reliable sentiment resource to be used for German text sentiment analysis. The created reliable resource called SentiWS ("SentimentWortschatz", Sentiment vocabulary) [27]. It incorporates sentiment bearing words with their positive or negative prior polarity score. In addition, in SentiWS a part of speech tag is assigned to each sentiment bearing word. In order to improve the sentiment bearing word coverage in SentiWS, a possible inflection forms for each sentiment bearing word have been added. Currently, SentiWS includes 1650 negative and 1818 positive words and its coverage is increased by including different word forms up to 16406 for positive and 16328 for negative words (after creation, the candidate word forms were manually examined). One of the resources used to create the SentiWS was based on 5100 positive and 5100 negative product reviews including 30074 and 36743 sentences, respectively. The polarity weighting is obtained based on the utilization of manually selected seed words that belong either to a positive or negative sentiment class. In order to infer the relationship between a seed word and its potential semantic orientation, a mutual information approach similar to Turney (2002) [30] has been used.

Pang et al. (2002) applied standard machine learning approaches e.g., Support Vector Machines (SVMs), Naïve Bayes (NB) and Maximum Entropy (ME), and achieved significant results on the sentiment classification task [25]. They found that the studied machine learning approaches are superior to the human-produced base line. They compared the performance of the three machine learning approaches and concluded that the results produced by the SVMs are slightly better compared to other approaches. Based on the achieved results, which were obtained based on different features used e.g., unigram, unigram+POS, adjectives, etc., they discovered that incorporating POS and n-gram models does not lead to any improvement compared to the simple unigram bag-of-words feature. Despite achieving good results, the performance of machine learning approaches in sentiment classification still lags behind the results achieved on standard traditional document classification using document topics [25]. Supervised machine learning approaches require human annotated data which is very laborious to obtain. Therefore, their applicability to tackle new domains is limited.

In order to benefit from lexical-based and machine learning approaches for sentiment analysis, Melville et al. (2009) proposed a combined approach, which augments lexical information obtained from a sentiment lexicon with a small set of labeled training examples, in the form of supervised learning [22]. Melville et al. (2009) demonstrated that lexical resource knowledge should not be discarded as it could reduce the effort of labeling large numbers of training examples, which can be used by the machine learning approach e.g., Naïve Bayes (NB) to detect the sentiment for new domains. Based on the obtained results, they found that the unified approach achieved better performance compared to the lexical-based or machine learning approaches used individually [22].

3. THE PROPOSED APPROACH

The aim of the algorithm is to automatically identify sentiment expressed in a consumer products review. This is achieved through four main steps: first, given a polar sentence, a sentence that bears positive or negative sentiment, only words that bear sentiment will be extracted. Earlier research in sentiment analysis e.g., [13] focused on extracting the semantic orientation of adjectives relying on the fact that adjectives can reflect most of the subjective content in a given text [14]. Isolated adjectives may be a good indicator for subjectivity but they sometimes lack sufficient context in order to infer the semantic orientation of the entire text [30]. Therefore, there is a need to disambiguate the ambiguous adjectives. Second, we infer the words' polarity in the test sentences through obtaining their statistical relationships, using pointwise mutual information (PMI), with training sentence polarity labels. Third, since negation plays an important role in sentiment where it reverses the original sentiment polarity of a word from one sentiment class into another, we checked if the sentiment polarity detection of a given word obtained in step 2 involved negations, if yes then its sentiment polarity will be flipped. Fourth, having the sentiment polarity value for each identified sentiment word in a test sentence, an accumulation of these sentiment polarity values will produce the semantic polarity for the entire product review.

3.1 The training Data

Boland et al. (2013) created a sentiment-annotated corpus of German Amazon products review [3]. It consists of sub corpora of reviews for different product types: books, mobiles, smartphones cameras, tablets and washing machine. In order to extract the reviews, they used a modified version of the Amazon reviews downloader and parser[1] and split the reviews into sentences using the ASV Segmentizer[2]. Each corpus of reviews for a product type was balanced to contain an equal number of sentences from positive and negative reviews (4-5 stars and 1-2 stars, respectively) and all sentences in reviews with a rating of 3 stars. All sentences in the resulting test sets where labeled as belonging to one of 4 categories with respect to the sentiment they expressed: positive, negative, mixed or neutral. This procedure resulted in a total of 63037 annotated sentences with 10500 of them being labeled by 3 or more annotators. Inter-rater agreement for these 10500 sentences reached a Fleiss' kappa of 0.6394 for book reviews, 0.7310 for washing machines, 0.6188 for cam-

[1]http://www.esuli.it/software/amazon-reviews-downloader-and-parser/

[2]http://asv.informatik.uni-leipzig.de

eras, 0.7013 for smartphones, 0.6690 for tablets and 0.7370 for mobiles.

For our evaluation, we selected a subset of only positive and negative annotations for each domain (See Table 1).

3.2 Pointwise Mutual Information (PMI)

Given a source of data, Pointwise Mutual Information (PMI) is a measure to calculate the correlation between two terms in a specific space e.g., corpora or web [1]. It was initially used by Church and Hanks (1990) to find similarity between two terms [9]. Large values of PMI indicate that two terms occur together more often and are semantically related. Small values of PMI indicate that one term is likely to appear when the other term is absent and therefore they are semantically not related.

Given an unlabeled set of sentences $T = \{t_1(w_1, .., w_n), \cdots, t_n(w_1, \cdots, w_n)\}$, where t_i denotes the ith test sentence and the w_i denotes the word i within it, our task is to automatically discover the sentiment polarity of each test sentence. Having manually annotated training sentences, that contain sentences with their sentiment polarities $S = \{s_1(w_1, .., w_n : lab), \cdots, s_n(w_1, .., w_n : lab)\}$, where s_i denotes the ith labeled training sentence, the w_i denotes the word ith within it and the lab refers to its sentiment polarity with either a positive or negative label. The correlation between each word in the test sentences and its sentiment polarity label found in the training set is computed as follows:

$$PMI(w, lab) = \sum_{i=1}^{n} \sum_{j=1}^{n} log_2 \frac{p(w_i, lab_j)}{p(w_i)p(lab_j)} \quad (1)$$

where $PMI(w, lab) = \begin{cases} + & if \ \ PMI(w, lab_+) > PMI(w, lab_-) \\ - & \text{otherwise.} \end{cases}$

The probability $p(w, lab)$ is estimated by counting how many times a word w_i is found with a given sentiment label (positive or negative) lab_j in the training sentences. The probabilities $p(w)$ and $p(lab)$ are estimated by counting the number of individual occurrences of each one independent from the other.

3.3 Ambiguous Adjectives Disambiguation

There are some adjective such as "high", "low", "small", "big" etc., that are not good indicators for sentiment polarity when they are used separately. Therefore, when such adjectives appear in the test sentence, it is necessary to automatically detect their sentiment polarities i.e., to discover if these ambiguous adjectives bear positive or negative sentiment. Even though the number of these ambiguous adjective is not abundant, they are used frequently to express views in opinionated texts.

Ambiguous adjective disambiguation has not been considered in depth in previous work for sentiment analysis [35] [20]. Nevertheless, few works has been done, for example, pattern-based method [34] and the SemEval-2010 task of disambiguating sentiment ambiguous adjectives (includes lexicon-based methods, machine learning methods (SVM classifier) etc.,) [33]). To our knowledge, we are the first to propose the conditional mutual information approach (See Section 3.3.1) to tackle the problem of ambiguous adjectives disambiguation in the sentiment analysis field.

In order to accomplish this task, these ambiguous adjectives must be known. Therefore, we used the ambiguous adjective list ("large", "many", "high", "thick", "deep", "heavy", "huge", "great", "small", "few", "low", "thin", "shallow", "light") obtained from [33]. For disambiguation, the accompanying ambiguous adjective noun can be a good indicator for the ambiguous adjective polarity classification. For clarification, we consider the following two examples, the sentence ("Der Kamera Bildschirm hat niedriger Auflösung", the camera display has low resolution) and the sentence ("Die Kamera überzeugt besonders durch den niedrigen Preis", the camera is particularly impressive because of the low price).

Based on the context in which it appears, the ambiguous adjective ("niedrig", low) can have a positive or negative polarity. The first example clearly expresses negative polarity where the noun "Auflösung - resolution" can be used for disambiguation while the second example expresses positive polarity where the noun ("Preis", price) can be used for disambiguation. In order to take different cases into account, depending on the presence of the target noun, we consider only nouns that are in close proximity to the target adjective within a context window of size $[-4, +4]$ (4 positions before and 4 positions after the target adjective). The context window of size $[-4, +4]$ refers to the fact that only nouns that are within the maximum distance of 4, before and after the target adjective, will be considered in an iterative process (the first closest noun found will lead to stop the iterative process).

In the previous examples, the disambiguation indicator noun is located on the right context window side of the ambiguous adjective. The distance between the disambiguation indicator noun (in both examples) and the ambiguous adjective is 1 $[0, 1]$. However, there are some cases where the ambiguous adjectives can be located with different distances from the ambiguous adjective. For clarification, we consider the following example ("aber der Kamera Preis war eindeutig zu hoch", but the camera's price was definitely too high). Here the context window will span since the disambiguation indicator noun is located on the left side context window with a distance of value 4 $[-4, 0]$. For identifying the negation, the window approach as discussed above is used. The subjective word must first be identified and its sentiment polarity score based on equation 1 is computed. For identifying the negation scope, a window based approach is used as discussed above in order to detect the negation markers (e.g., "nicht", not, "nie", never etc.,) position around the negated subjective words. After extracting the subjective word with its negation marker if applicable, the polarity of the subjective word is then flipped.

3.3.1 Conditional Mutual Information (CMI)

Given two random variables, pointwise mutual information is a statistical approach used to reduce the uncertainty of one random variable based on the knowledge of the other. However, in the case of ambiguous adjectives, pointwise mutual information cannot be used as a third value is needed for adjective disambiguation which is the accompanying adjective noun. Therefore, given three random variables X, Y and Z, our task is to reduce the uncertainty of X due to the knowledge of Y when Z is given. For simplicity, X is denoting the target word w, Y is the given polarity label and Z is the accompanying adjective noun.

$$CMI(X;Y|Z) = \sum_{x\in X}\sum_{y\in Y}\sum_{z\in Z} p(x,y,z) log \frac{p(x;y|z)}{p(x|z)p(y|z)} \quad (2)$$

The conditional probability $p(x;y|z) = \frac{p(x,y,z)}{p(z)}$, $p(x|z) = \frac{p(x,z)}{p(z)}$ and $p(y|z) = \frac{p(y,z)}{p(z)}$

based on the previousely given values, quation 1 can be written as:

$$CMI(X;Y|Z) = \sum_{x\in X}\sum_{y\in Y}\sum_{z\in Z} p(x,y,z) log \frac{p(z)p(x,y,z)}{p(x,z)p(y,z)} \quad (3)$$

the probabilities $p(x,y,z)$, $p(x,z), p(y,z)$ and $p(z)$ is computed same as in equation 1.

Algorithm 1 shows the main abstract steps of the proposed approach in how it perform the sentiment classification task.

4. EVALUATION

In our evaluation, we used the k-fold cross validation method [16] in that the data is split into k folds (usually from 5-10 folds), where k-1 folds is used for training the algorithm and the remaining one is used for testing the algorithm. A special case of k-fold cross validation is leave-one-out cross validation (LOOCV) where k is equal to the size of the data. In LOOCV, iteration is repeated k times over the entire training data where the first instance is taken out for testing and the rest k-1 is used for training then the second instance is taken out for testing and the rest of k-1 folds for training and so on. LOOCV is widely used when the available data is scarce [6] where the LOOCV process prevents wasting any data and accuracy estimation gained based on using LOOCV is known to be unbiased [11] [7].

For our evaluation, we used 6 diverse test sets obtained from [3] (See Section 3.1). These 6 domains are books, camera, mobile, smartphone, tablets and washing machine. Table 1 shows the characteristics of the test data including the diverse domains with their corresponding test sentence numbers.

Domain	Positive	Negative	Total
Books	3705	4557	8262
Mobile	774	887	1661
Smartphone	2211	3088	5299
Camera	490	398	888
Tablets	2452	3112	5564
Washing Machine	1153	1332	2485
Total	**10785**	**13374**	**24159**

Table 1: Number of the test sentences across the 6 domains.

4.1 True Error Rate

The true error rate based on using the LOOCV method is computed in the same way as it is defined in [8] [15]. Given a test set T, where $T = \{t_1, t_2, ..., t_n\}$, the true error E, where $E = \{e_1, e_2, ..., e_n\}$ is estimated as the average error rate obtained from testing the entire test set T. The true error rate E is estimated based on the following equation:

Algorithm 1 Algorithm for sentiment polarity classification

- **Input:**
  A given test sentence t_m with its POS tag features $t_m \leftarrow \{w_{1_{pos}}, .., w_{n_{pos}}\}$,
  A given training set $S \leftarrow \{s_1(w_1, ..., w_n : label), ..., s_n(w_1, ..., w_n : label)\}$

- **Outputs:**
  The test sentence t_m with its assigned polarity label $t_m \leftarrow \{w_1, ..., w_n : label\}$

# compute the t_m sentiment score for + and − class
for $i \leftarrow 0$ to 1 **do**
  if $(i \leftarrow 0)$ **then**
    $Label \leftarrow +$
  else
    $Label \leftarrow -$
  end if
  for each *word* $w \in t_m$ **do**
    $Adjectives \leftarrow getAdj(w)$; # identify a sentiment word
  end forend for
# disambiguate ambiguous adjectives based on the accompanying noun
  for each *word* $w \in Adjectives$ **do**
# update adjectives list, $k \leftarrow \pm 4$ "context window size"
    $Adjectives \leftarrow disambiguate(w, t_m, k)$
# compute individual sentiment score for each $w \in Adjectives$, loop over all training sentences
    for $j \leftarrow 0$ to $|S|$ **do**
      $score \leftarrow getScore(w \in adjectives, Label)$ based on Eq. 1
Label either + or − , (See step 2)
      $scoreLabel \leftarrow scoreLabel + score$;
    end forend for
negation detection
    $negation \leftarrow getNeg(w, t_m, k)$
    if (*negated*) **then**
      $scoreLabel \leftarrow \neg scoreLabel$
    end if
# compute sentence sentiment accumulation score
    $t_m scoreLabel \leftarrow t_m scoreLabel + scoreLabel$
  end forend for
end forend for
if ($t_m$score+ $> t_m$score−) **then**
  $classification \leftarrow -$
else
  $classification \leftarrow +$
end if
return $classification$

$$E = \frac{1}{|T|} \sum_{i=1}^{|E|} e_i \quad (4)$$

Out of the true error rate E, the overall accuracy can be calculated.

In order to evaluate the performance of the proposed algorithm, we implemented all parts of the algorithm discussed in Section 3.2 and Section 3.3. The implementation of the entire algorithm was java based. In order to efficiently perform the training and the evaluation; the training data has been indexed by the Apache Lucene library[3] so all statistical information needed by the algorithm can be efficiently obtained. In order to increase the performance of the algorithm, it was necessary to perform a lemmatization. We used the lemmatization features included in the Apache Lucene library. Grammatical features (adjective, noun etc.,) are important features which are used for subjectivity detection and disambiguation. Therefore, to obtain those grammatical features, we used the TreeTagger [28].

4.2 Experiments

The main experiment we performed aimed to assess the accuracy of predicting the correct sentence polarity across the entire corpus for the 6 different domains. Here, we examine the performance of the proposed algorithm (based on Pointwise Mutual Information (PMI)) against the performance of the German sentiment dictionary SentiWS [27] (See Section 2). The result achieved by the SentiWS was used as a baseline. As is the case when using the dictionary-based approach for sentiment analysis (See Section 2), we obtained the sentiment for each test sentence by simply accumulating the positive and the negative words included in the test sentence. The greater overall polarity, either positive or negative, will reflect the given sentence's polarity i.e., more opinion positive words indicates a positive sentence and negative words implies a negative sentence.

To evaluate whether the algorithm improvement (tackling the ambiguous adjective issue, we name this part of the algorithm as Revised Mutual Information RMI) is useful and has led to an improvement in accuracy, we added a second experiment where we performed accuracy comparisons between the dictionary-based approach, PMI approach and the RMI approach. Table 2 and Figure 1 demonstrates the different accuracy results achieved by the three approaches. As is shown in Table 2 and Figure 1, overall, the RMI approach dominates in almost all domains, while the dictionary-based approach is the worst of five and is only better in one (camera domain) domain. The proposed algorithm low performance for camera's domain was due to the much-reduced training corpus size in the camera domain (888 training sentences; 490 of them are positive and 398 of them are negative) compared to the other domains and hence this led to a reduction in the proposed approach performance, as no significant data for some test sentences in this domain were obtainable.

As is shown in Table 2, the accuracy rate for the proposed algorithm achieved more than 73.68%, on average, across all domains, while the dictionary-based approach reached an average of 70.63% across all domains. On the other hand, the accuracy rate for the proposed approach gradually improved due to ambiguous adjective disambiguation and attained more than 75.27% in comparison to other approaches. The RMI algorithm performance could be clearly increased if a smaller sized context window is used. However, the context window size used in our experiment was 4 (See Section 3.3) in order to cover as many possible noun positions around the ambiguous adjective. In some cases, the larger context window size led to a wrong noun selection which in turn led to accuracy reduction.

In our future work, the accompanying ambiguous adjective/noun selection process will be improved in order to increase the overall algorithm accuracy. The algorithm has been tested based only on adjectives. However, there are also nouns, adverbs etc., which bear sentiment and should be used. Furthermore, we will increase our test to cover nouns, adverbs, etc., however, there is urgent need for careful selection of nouns as most nouns bear no sentiment and using them without a filter could clearly cause noise in the achieved result. Another reason of the performance reduction is that there is noise in the training data. For example, often the training sentences contain only one word, one expression or symbol e.g., ("neeeeeeeee", nooooooo) was annotated as negative by the annotators. However, the algorithm has no clue in distinguishing the correct sentiment when the words are used incorrectly or have no significant statistical score in the training data. Furthermore, symbols such as ":)" or ":(" have been frequently used which have not been covered by the algorithm. Another reason of the performance reduction is that the algorithm accuracy depends also in the POS tagger accuracy which is not free of errors. Possible investigation in the future work is to use different POS tagger for German text and examine their effect in the sentiment classification algorithm.

Nevertheless, the results achieved indicate that our algorithm achieved significant agreement with the annotation done by experts. The proposed algorithm showed stable performance across all domains except the camera domain due to the reduced camera corpus size. The algorithm has been tested in 24159 test sentences across the six domains and showed an overall accuracy of 75.27% which we regard as promising results in the difficult sentiment classification task.

Domains	**SentiWS**	**PMI**	**Revised MI**
Books	67.67%	72.73%	73.08%
Mobile	74.61%	77.63%	81.01%
Smartphone	70.28%	76.48%	77.52 %
Camera	72.63%	63.41%	67.09%
Tablets	69.57%	73.17%	74.08%
Washing Machine	69.02%%	78.63%	78.83%
Accuracy	**70.63%**	**73.68%**	**75.27%**

Table 2: Average accuracy in % of SentiWS, Pointwise Mutual Information (PMI) and Revised MI.

5. CONCLUSION

In this paper, we described different state-of-the art approaches in the sentiment classification task. We presented an approach that attempted to classify the sentiment represented by sentences in product reviews. Based on experiments which have been performed on a large test corpus

[3]http://lucene.apache.org/

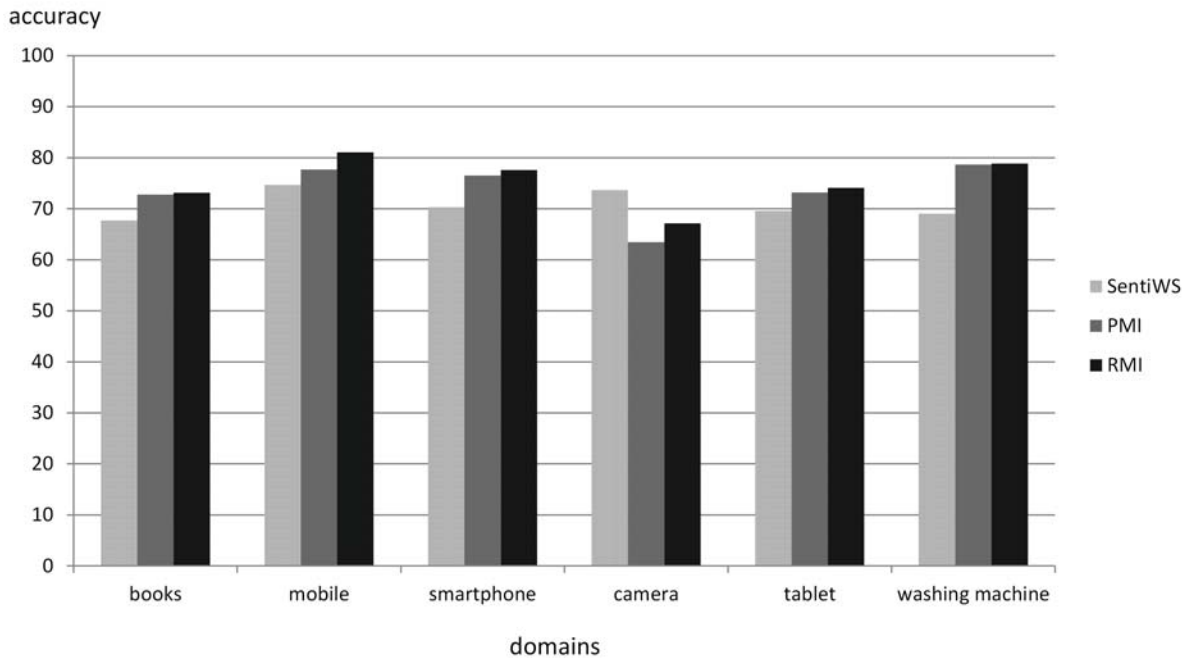

Figure 1: The obtained accuracy for each domain.

across 6 different domains, the proposed approach shows clear performance compared to the dictionary-based approach. Furthermore, the revised MI algorithm (RMI) resulted in improvements of the PMI algorithm, taking into account the disambiguation of the ambiguous adjectives. In our ongoing research, one of our interests is to incorporate more opinionated words such as noun, adverb, etc., rather than only adjectives. Since including nouns, adverbs, etc., can lead to significant noise in the results, we are currently developing a solid mechanism for improving the selection of only the opinionated nouns and adverbs that bear sentiment.

6. ACKNOWLEDGMENTS

This work was supported by the German Federal Ministry of Education and Research (BMBF), under Grant No. 01UA1101C (eTRACES Project).

7. REFERENCES

[1] F. Ahmed, A. Nürnberger, and M. Nitsche. Supporting arabic cross-lingual retrieval using contextual information. In A. Rauber and A. de Vries (Eds.), editors, *Multidisciplinary Information Retrieval*, volume 6653, pages 30–45. Springer-Verlag, Berlin-Heidelberg, 2011.

[2] E. M. Airoldi, X. Bai, and R. Padman. Markov blankets and meta-heuristic search: Sentiment extraction from unstructured text. *Lecture Notes in Computer Science*, 3932 (Advances in Web Mining and Web Usage Analysis):167–187, 2006.

[3] K. Boland, A. Wira-Alam, and R. Messerschmidt. Creating an annotated corpus for sentiment analysis of german product reviews. Technical report, GESIS - Leibniz Institute for the Social Sciences, 2013. http://www.gesis.org/en/publications/gesis-technical-reports/.

[4] E. Cambria, C. Havasi, and A. Hussain. Senticnet 2: A semantic and affective resource for opinion mining and sentiment analysis. In *In Proceedings of the FLAIRS Conference*, 2012.

[5] E. Cambria and A. Hussain. *Sentic Computing: Techniques, Tools, and Applications*. Springer, Dordrecht, Netherlands, 2012. Book Link: http://www.springer.com/biomed/book/978-94-007-5069-2.

[6] G. C. Cawley and N. L. Talbot. Efficient approximate leave-one-out cross-validation for kernel logistic regression. *Mach. Learn.*, 71(2-3):243–264, 2008.

[7] G. C. Cawley and N. L. C. Talbot. Fast exact leave-one-out cross-validation of sparse least-squares support vector machines. *Neural Networks*, 17(10):1467–1475, 2004.

[8] O. Chapelle, V. Vapnik, O. Bousquet, and S. Mukherjee. Choosing multiple parameters for support vector machines. *Mach. Learn.*, 46(1-3):131–159, 2002.

[9] K. W. Church and P. Hanks. Word association norms, mutual information, and lexicography. *Comput. Linguist.*, 16(1):22–29, 1990.

[10] G. Druck, G. Mann, and A. McCallum. Learning from labeled features using generalized expectation criteria. In *Proceedings of the 31st annual international ACM SIGIR conference on Research and development in information retrieval*, SIGIR '08, pages 595–602, 2008.

[11] B. Efron. Estimating the error rate of a prediction rule: Improvement on cross-validation. *Journal of the*

American Statistical Association, 78(382):316–331, 1983.

[12] A. Esuli and F. Sebastiani. Sentiwordnet: A publicly available lexical resource for opinion mining. In *Proceedings of LREC-2006, The fifth international conference on Language Resources and Evaluation*, pages 62–66, 2006.

[13] V. Hatzivassiloglou and J. M. Wiebe. Effects of adjective orientation and gradability on sentence subjectivity. In *Proceedings of the 18th conference on Computational linguistics - Volume 1*, COLING '00, pages 299–305, Stroudsburg, PA, USA, 2000. Association for Computational Linguistics.

[14] M. Hu and B. Liu. Mining and summarizing customer reviews. In *Proceedings of the tenth ACM SIGKDD international conference on Knowledge discovery and data mining*, KDD '04, pages 168–177, 2004.

[15] S. Kale, R. Kumar, and S. Vassilvitskii. Cross-validation and mean-square stability. In *Proceedings of the Second Symposium on Innovations in Computer Science (ICS2011)*, pages 487–495, 2011.

[16] R. Kohavi. A study of cross-validation and bootstrap for accuracy estimation and model selection. In *Proceedings of the 14th international joint conference on Artificial intelligence - Volume 2*, IJCAI'95, pages 1137–1143, 1995.

[17] C. W. K. Leung, S. C. F. Chan, and F. Chung. Integrating Collaborative Filtering and Sentiment Analysis: A Rating Inference Approach. In *Proceedings of The ECAI 2006 Workshop on Recommender Systems*, pages 62–66, 2006.

[18] Y. Lin, J. Zhang, X. Wang, and A. Zhou. An information theoretic approach to sentiment polarity classification. In *Proceedings of the 2nd Joint WICOW/AIRWeb Workshop on Web Quality*, WebQuality '12, pages 35–40, 2012.

[19] B. Liu. Opinion observer: Analyzing and comparing opinions on the web. In *In WWW '05: Proceedings of the 14th international conference on World Wide Web*, pages 342–351. ACM Press, 2005.

[20] B. Lu and B. K. Tsou. Cityu-dac: Disambiguating sentiment-ambiguous adjectives within context. In *Proceedings of the 5th International Workshop on Semantic Evaluation*, pages 292–295, July 2010.

[21] Q. Mei, X. Ling, M. Wondra, H. Su, and C. Zhai. Topic sentiment mixture: modeling facets and opinions in weblogs. In *Proceedings of the 16th international conference on World Wide Web*, WWW '07, pages 171–180, New York, NY, USA, 2007. ACM.

[22] P. Melville, W. Gryc, and R. D. Lawrence. Sentiment analysis of blogs by combining lexical knowledge with text classification. In *Proceedings of the 15th ACM SIGKDD international conference on Knowledge discovery and data mining*, KDD '09, pages 1275–1284, New York, NY, USA, 2009. ACM.

[23] G. A. Miller. Wordnet: a lexical database for english. *Communications of the ACM*, 38(11):39–41, 1995.

[24] K. Nigam, J. Lafferty, and A. McCallum. Using maximum entropy for text classification. In *Proceedings of the IJCAI-99 Workshop on Machine Learning for Information Filtering*, pages 61–67, 1999.

[25] B. Pang, L. Lee, and S. Vaithyanathan. Thumbs up?: sentiment classification using machine learning techniques. In *Proceedings of the ACL-02 conference on Empirical methods in natural language processing - Volume 10*, EMNLP '02, pages 79–86, Stroudsburg, PA, USA, 2002. Association for Computational Linguistics.

[26] G. Qiu, B. Liu, J. Bu, and C. Chen. Expanding domain sentiment lexicon through double propagation. In *Proceedings of the 21st international jont conference on Artifical intelligence*, IJCAI'09, pages 1199–1204, 2009.

[27] R. Remus, U. Quasthoff, and G. Heyer. Sentiws - a publicly available german-language resource for sentiment analysis. In N. C. C. Chair), K. Choukri, B. Maegaard, J. Mariani, J. Odijk, S. Piperidis, M. Rosner, and D. Tapias, editors, *Proceedings of the Seventh International Conference on Language Resources and Evaluation (LREC'10)*, Valletta, Malta, may 2010. European Language Resources Association (ELRA).

[28] H. Schmid. Improvements in part-of-speech tagging with an application to german. In *In Proceedings of the ACL SIGDAT-Workshop*, pages 47–50, 1995.

[29] R. M. Tong. An operational system for detecting and tracking opinions in on-line discussions. In *Proceedings of the ACM SIGIR 2001 Workshop on Operational Text Classification*, pages 1–6, 2001.

[30] P. D. Turney. Thumbs up or thumbs down?: semantic orientation applied to unsupervised classification of reviews. In *Proceedings of the 40th Annual Meeting on Association for Computational Linguistics*, ACL '02, pages 417–424, 2002.

[31] R. Valitutti. Wordnet-affect: an affective extension of wordnet. In *In Proceedings of the 4th International Conference on Language Resources and Evaluation*, pages 1083–1086, 2004.

[32] T. Wilson, J. Wiebe, and R. Hwa. Just how mad are you? finding strong and weak opinion clauses. In *Proceedings of the 19th national conference on Artificial intelligence*, AAAI'04, pages 761–767, 2004.

[33] Y. Wu and P. Jin. Semeval-2010 task 18: Disambiguating sentiment ambiguous adjectives. In *Proceedings of the 5th International Workshop on Semantic Evaluation*, pages 81–85, Uppsala, Sweden, July 2010. Association for Computational Linguistics.

[34] Y. Wu, M. Wang, P. Jin, and S. Yu. Disambiguate sentiment ambiguous adjectives. In *Proceedings of the IEEE International Conference on Natural Language Processing and Knowledge Engineering*, pages 1191–1199, 2008.

[35] S.-C. Yang and M.-J. Liu. Ysc-dsaa: An approach to disambiguate sentiment ambiguous adjectives based on saaol. In *Proceedings of the 5th International Workshop on Semantic Evaluation*, pages 440–443, July 2010.

[36] H. Yu and V. Hatzivassiloglou. Towards answering opinion questions: separating facts from opinions and identifying the polarity of opinion sentences. In *Proceedings of the 2003 conference on Empirical methods in natural language processing*, EMNLP '03, pages 129–136, 2003.