Knowledge-Based Systems 52 (2013) 201–213

Contents lists available at ScienceDirect

Knowledge-Based Systems

journal homepage: www.elsevier.com/locate/knosys

Care more about customers: Unsupervised domain-independent aspect detection for sentiment analysis of customer reviews

Ayoub Bagheri [a,*], Mohamad Saraee [b], Franciska de Jong [c]

[a] Intelligent Database, Data Mining and Bioinformatics Lab, Electrical and Computer Engineering Department, Isfahan University of Technology, Isfahan, Iran
[b] School of Computing, Science and Engineering, University of Salford, Manchester, UK
[c] Human Media Interaction Group, University of Twente, Enschede, Eramus Universiteit Rotterdam, Erasmus Studio, P.O. Box 1738, 3000 DR Rotterdam, The Netherlands

ARTICLE INFO

Article history:
Received 8 January 2013
Received in revised form 21 July 2013
Accepted 2 August 2013
Available online 13 August 2013

Keywords:
Aspect detection
Opinion mining
Review mining
Sentiment analysis
Implicit aspect

ABSTRACT

With the rapid growth of user-generated content on the internet, automatic sentiment analysis of online customer reviews has become a hot research topic recently, but due to variety and wide range of products and services being reviewed on the internet, the supervised and domain-specific models are often not practical. As the number of reviews expands, it is essential to develop an efficient sentiment analysis model that is capable of extracting product aspects and determining the sentiments for these aspects. In this paper, we propose a novel unsupervised and domain-independent model for detecting explicit and implicit aspects in reviews for sentiment analysis. In the model, first a generalized method is proposed to learn multi-word aspects and then a set of heuristic rules is employed to take into account the influence of an opinion word on detecting the aspect. Second a new metric based on mutual information and aspect frequency is proposed to score aspects with a new bootstrapping iterative algorithm. The presented bootstrapping algorithm works with an unsupervised seed set. Third, two pruning methods based on the relations between aspects in reviews are presented to remove incorrect aspects. Finally the model employs an approach which uses explicit aspects and opinion words to identify implicit aspects. Utilizing extracted polarity lexicon, the approach maps each opinion word in the lexicon to the set of pre-extracted explicit aspects with a co-occurrence metric. The proposed model was evaluated on a collection of English product review datasets. The model does not require any labeled training data and it can be easily applied to other languages or other domains such as movie reviews. Experimental results show considerable improvements of our model over conventional techniques including unsupervised and supervised approaches.

1. Introduction

With the rapid growth of user-generated content on the internet, the number of customer reviews that a product or service receives grows rapidly. A significant number of websites, blogs and forums (e.g., www.amazon.com, rottentomatoes.com, epinions.com) allow customers to post opinions about a variety of products or services. This online word of mouth behavior introduces a new and important source of information for business intelligence and marketing. In the other words customer reviews are essential to other potential customers, retailers and product manufacturers (potential users) in their efforts to understand the general opinions of customers and help them to make better decisions. As the number of customer reviews expands, it becomes very hard for users to obtain a comprehensive view of opinions of previous customers about various aspects of products through a manual analysis. Consequently proper analysis and summarization of customer reviews can further enable potential users to visualize previous positive and negative opinions about specific features or aspects of products. Therefore it is highly desirable to produce an automatic analysis or summary of customer reviews.

For the past few years, sentiment analysis (or opinion mining) for online customer reviews has attracted a great deal of attentions from researchers of data mining and natural language processing [1,3,5,7,8,11,9,24,25,27,33].

Sentiment analysis is a type of text analysis under the broad area of text mining and computational intelligence. Three fundamental problems in sentiment analysis are: aspect detection, opinion word detection and sentiment orientation identification [24,27,33].

Aspects are topics on which opinions are expressed. In the field of sentiment analysis, other names for aspect are: features, product features or opinion targets [3,5,7,8,6,12,24,27,33]. Aspects are important because without knowing them, the opinions expressed

[*] Corresponding author. Tel.: +98 311 3912450.
E-mail addresses: a.bagheri@ec.iut.ac.ir, ayoub.bagheri@gmail.com (A. Bagheri), m.saraee@salford.ac.uk (M. Saraee), f.m.g.dejong@utwente.nl (F. de Jong).

0950-7051/$ - see front matter © 2013 Elsevier B.V. All rights reserved.
http://dx.doi.org/10.1016/j.knosys.2013.08.011

in a sentence or a review are of limited use. For example, in the review sentence "after using iPod, I found the size to be perfect for carrying in a pocket", "size" is the aspect for which an opinion is expressed. Likewise aspect detection is critical to sentiment analysis, because its effectiveness dramatically affects the performance of opinion word detection and sentiment orientation identification. Therefore, in this study we concentrate on aspect detection for sentiment analysis.

Existing aspect detection methods can broadly be classified into two major approaches: supervised and unsupervised. Supervised aspect detection approaches require a set of pre-labeled training data. Although the supervised approaches can achieve reasonable effectiveness, building sufficient labeled data is often expensive and needs much human labor. Since unlabeled data are generally publicly available, it is desirable to develop models that work with unlabeled data. Additionally, due to variety and wide range of products and services being reviewed on the internet, supervised, domain-specific or language-dependent models are often hard to apply. Therefore we conclude the framework for the aspect detection must be robust and easily transferable between domains or languages.

In this paper, we present a novel unsupervised model which addresses the core tasks necessary to detect explicit and implicit aspects from review sentences in a sentiment analysis system. Our model differs from existing techniques in that it requires no labeled training data or additional information, not even for the initial seed information. Therefore the model can easily be transferred between domains or languages. The proposed model is based on the observation that there is inter-relation information between the aspects in reviews. Inter-relation information is the probability of the co-occurrence of two aspects in a review. Therefore the model explores review dataset by using both frequency-based and inter-relation information to find the aspects. Furthermore we have found that opinion words and aspects themselves have relations in opinionated sentences. Finally the model uses explicit extracted aspects and opinion words to detect implicit aspects.

In the remainder of this paper, Section 2 gives a definition of the aspect-level sentiment analysis, detailed discussions of existing works on aspect detection will be given in Section 3. Section 4 describes the proposed aspect detection model for sentiment analysis, including the overall process and specific aspects of the design of the workflow. Subsequently we describe our empirical evaluation and discuss the major experimental results in Section 5. Finally we conclude with a summary and some future research directions in Section 6.

2. Aspect-level sentiment analysis

Opinions can be expressed about anything, e.g., a topic, a product, a service, an individual, an event, an organization or any attributes of them. Hence we use the notation of *aspect* to denote the target object that has been evaluated. An opinion (as expressed by means of opinion words) is a positive or negative sentiment, attitude, emotion or appraisal about an aspect. Positive and negative are called sentiment or opinion orientations [10,6]. In general there are two types of reviews: standard and ironic reviews. Ironic review refers to the specific case of text where a sentence or expression with prior positive polarity is figuratively used for expressing a negative opinion. Ironic expressions are recognized in literature as a specific phenomenon that can harm sentiment analysis systems [2,18,19].

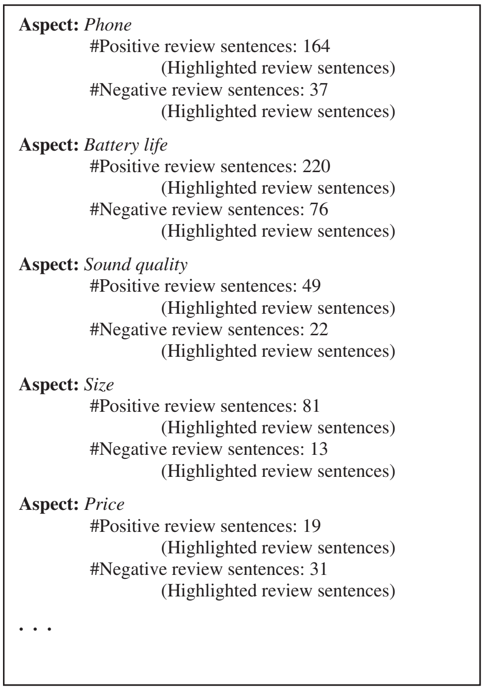
Sentiment analysis at the document-level or at the sentence-level is useful in many applications, but it does not provide the necessary detail needed for many use cases. A positive document or sentence sentiment for a certain object does not mean that the author has positive opinions on all aspects of the object. Likewise, a negative document or sentence sentiment does not mean that the author dislikes everything. In a document or sentence, the author may describe both positive and negative aspects of the product object, although the general sentiment on the object may be either positive or negative. Document and sentence sentiment analysis does not provide such information. To obtain these details we need to switch to a more fine-grained level of analysis and apply the aspect level sentiment analysis. Most aspect-level sentiment analysis approaches require the availability of opinions from a large number of users [10,31]. For the feasibility therefore, some form of summary of opinions is desirable. A common approach is called aspect-based opinion summary [10]. Figs. 1 and 2 represent two types of illustration of aspect-based opinion summarization.

Figs. 1 and 2 are illustrating two examples of aspect-based sentiment summary modeling. The models summarize all the reviews of a particular cellphone. For each model we can see which aspects have been taken into account, plus the number of positive and negative review sentences. In Fig. 1 we have highlighted individual review sentences additionally.

In aspect-level sentiment analysis there are two types of aspects: explicit and implicit. For example, in the sentence, "The signal quality of this phone is amazing", the aspect is "signal quality" of the object represented by "this phone". In this example the aspect is an explicit aspect as it appears in the sentence. In the sentence, "This phone is too light", the aspect is "weight" which is an implicit aspect as it does not appear in the sentence but it is implied. In the proposed model in this paper we study the problem of detecting explicit and implicit aspects in sentiment analysis for standard and ironic reviews.

3. Related works

Several methods have been proposed, mainly in the context of product review mining in a broad range of study fields, from document to aspect level sentiment analysis for standard, ironic or


Aspect: *Phone*
#Positive review sentences: 164
(Highlighted review sentences)
#Negative review sentences: 37
(Highlighted review sentences)

Aspect: *Battery life*
#Positive review sentences: 220
(Highlighted review sentences)
#Negative review sentences: 76
(Highlighted review sentences)

Aspect: *Sound quality*
#Positive review sentences: 49
(Highlighted review sentences)
#Negative review sentences: 22
(Highlighted review sentences)

Aspect: *Size*
#Positive review sentences: 81
(Highlighted review sentences)
#Negative review sentences: 13
(Highlighted review sentences)

Aspect: *Price*
#Positive review sentences: 19
(Highlighted review sentences)
#Negative review sentences: 31
(Highlighted review sentences)

. . .

Fig. 1. Example 1 of an aspect-based sentiment summary.

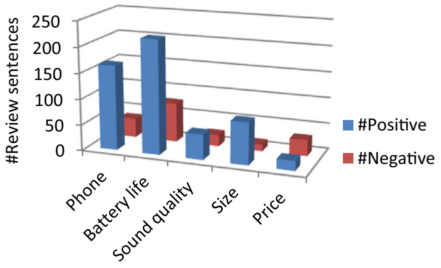


Fig. 2. Example 2 of an aspect-based sentiment summary.

spam reviews [3,7,8,6,12,16,21,27,33,18,19]. In the review mining task, aspects usually refer to opinion targets and product features, which are defined as product components or attributes. Existing aspect and product feature extraction techniques use both supervised and unsupervised methods for finding explicit and implicit aspects from a review [3,5,7,14,16,17,21–24,27,29,33–35].

The earliest attempt at aspect detection was based on the classic information extraction approach of using frequently occurring noun phrases presented by Hu and Liu [7]. Their work can be considered as the initiator work on aspect extraction from reviews. They use association rule mining (ARM) based on the Apriori algorithm to extract frequent itemsets as explicit product features, only in the form of noun phrases. In association rule mining, the algorithm does not consider the position of the words in a sentence, therefore in order to remove incorrect frequent features, they use feature pruning that consists of compactness pruning and redundancy pruning. Their approach works well in detecting aspects that are strongly associated with a single noun, but are less useful when aspects encompass many low-frequency terms. Proposed model in our study works well with low-frequency terms and uses heuristic POS patterns to extract the candidates for aspect. In addition to the frequency-based information, our model uses inter-relation information between the aspects.

Popescu and Etzioni [16] developed an unsupervised information extraction system called OPINE. Given a particular product and a corresponding set of reviews, OPINE first extracts noun phrases from reviews and retains those with frequency greater than an experimentally set threshold and then assesses those by OPINE's assessor for extracting explicit aspects. The assessor evaluates a noun phrase by computing a Point-wise Mutual Information (PMI) score between the phrase and meronymy discriminators associated with the product class. OPINE outputs a set of product aspects, each accompanied by a list of associated opinions which are ranked based on strength. Our approach differs from OPINE in the representation and construction of extraction patterns and in the measurements of extracted aspects.

Yi et al. [29] developed a set of aspect candidate extraction heuristics for extracting an aspect from product reviews. They introduced an aspect as a part of relationship with the given topic, an attribute of relationship with the given topic, and an attribute of relationship with a known aspect of the given topic. Based on the observation that aspect terms are nouns, they extract only noun phrases from documents and apply two feature selection algorithms, mixture language model [32] and likelihood ratio [4]. The process of candidate generation in our study is similar to the Yi et al. approach, but our proposed patterns include more candidates for the aspects.

Somprasertsri and Lalitrojwong's [21] proposed a supervised model for aspect detection by combining lexical and syntactic features with a maximum entropy technique. They defined four different features for learning the maximum entropy: Aspects and their POS tags, Rare words, Alphanumeric feature and Dependency from syntactic parse tree. They extracted the learning features from an annotated corpus. Their approach uses a maximum entropy classifier for extracting aspects and includes the postprocessing step to discover the remaining aspects in the reviews by matching the list of extracted aspects against each word in the reviews. We use Somprasertsri and Lalitrojwong's work for a comparison to our proposed model, because the model in our study is completely unsupervised.

Wei et al. [27] proposed a semantic-based product aspect extraction (SPE) method. The SPE technique exploits a list of positive and negative adjectives defined in the General Inquirer to recognize opinion words semantically and subsequently extract product features (aspects) expressed in customer reviews. Their approach begins with preprocessing task, and then employs the association rule mining to identify candidate product aspects. The SPE technique employs the same pruning rules as proposed in [7] in the pruning step to produce frequent product aspects from the set of candidate aspects. Afterward, on the basis of the list of positive and negative adjectives defined in the General Inquirer, the semantic-based refinement step identifies and then removes from the set of frequent aspects possible non-product aspects and opinion-irrelevant product aspects. In addition, the semantic-based refinement step attempts to discover infrequent product aspects and adds them to the list of product aspects extracted. The SPE approach relies primarily on frequency- and semantic-based extraction for the aspect detection, but in our study we use frequency-based and inter-relation information between the aspects and give more importance to multi-word aspects and aspects with an opinion word in the review sentence. Additionally in our model we extract implicit aspects from review sentences.

Finally, Zhu et al. [35] developed an aspect-based unsupervised opinion polling system. In their work, a multi-aspect bootstrapping method based on RlogF metric [20] and an ambiguity degree is proposed to learn aspect-related terms for each aspect to be used for aspect identification. RlogF metric for each aspect candidate t can be computed by: $f(t, T) * R(t, T)$, where $f(t, T)$ is the frequency of co-occurrence of candidate aspect t and current seed set T, and $R(t, T)$ is the ratio of $f(t, T)$ over frequency of t. By calculating RlogF, they rank aspect candidates based on the RlogF values, and then they propose three scores: rank-based score, ambiguity degree and importance score. Zhu et al. define the rank-based score based on the extracted ranks of RlogF values. In their approach they prefer single-word aspects over multi-word aspects. Therefore they proposed an ambiguity degree based on the RlogF metric to penalize the multi-word aspects. Finally they defined an importance socre for each candidate aspect by combining rank-based and ambiguity degree scores. In addition they use an aspect-based segmentation model by introducing a criterion function to segment a multi-aspect sentence into multiple single-aspect units as basic units for opinion polling. There are two differences between their approach and ours: (i) they use supervised seed information while in our approach we automatically select the seed set, and (ii) instead of RlogF for scoring each candidate aspect, we propose method, named A-score metric, uses both frequency-based and inter-relation information of words to score each candidate aspect. In our study we conduct experiments using unsupervised seed information could achieve favorable performance.

Our work on aspect detection is designed with the aim to minimize the role of supervision, so as to maximize the possibility to use it across different types of domains, as well as across languages. The motivation is to build a model to work on the characteristics of the words in reviews and inter-relation information between them, and to take into account the influence of an opinion word on detecting the explicit or implicit aspect.

4. Aspect detection model for sentiment analysis

Fig. 3 gives the architectural overview of the proposed model used for detecting explicit and implicit aspects in sentiment analysis. The basic hypotheses in this model are about using frequency-based and inter-relation information of the aspects together, employing the influence of an opinion word in the review sentence and giving more importance to multi-word aspects. This model proves using these hypotheses all together attain to highly effective results for product aspect extraction.

The model performs the detection in four main steps: finding multi-word aspects and using heuristic rules, employing proposed iterative bootstrapping with A-score metric, aspect pruning and detecting implicit aspects. The input to the model is a dataset of reviews and the output is a set of aspects from the reviews. Firstly, the model segments the reviews into sentences, then for each sentence POS tagging is performed, and candidates for aspects and words corresponding to the polarity lexicon are extracted. Then a stemming step is used for each aspect candidate, and single- and multi-word aspects are extracted by utilizing a generalized statistical measure. Some heuristic rules are applied to filter less informative aspects. Then a bootstrapping algorithm is employed, based on a newly-defined metric and an unsupervised initial seed set, to find aspects with the highest information. Not all aspects detected by the bootstrapping algorithm are genuine and useful aspects. There are also some redundant ones. Therefore the model uses aspect pruning to remove these incorrect aspects. Finally a new graph-based approach for extracting implicit aspect is introduced.

Below, we discuss each of the functions in the proposed aspect detection model of Fig. 3.

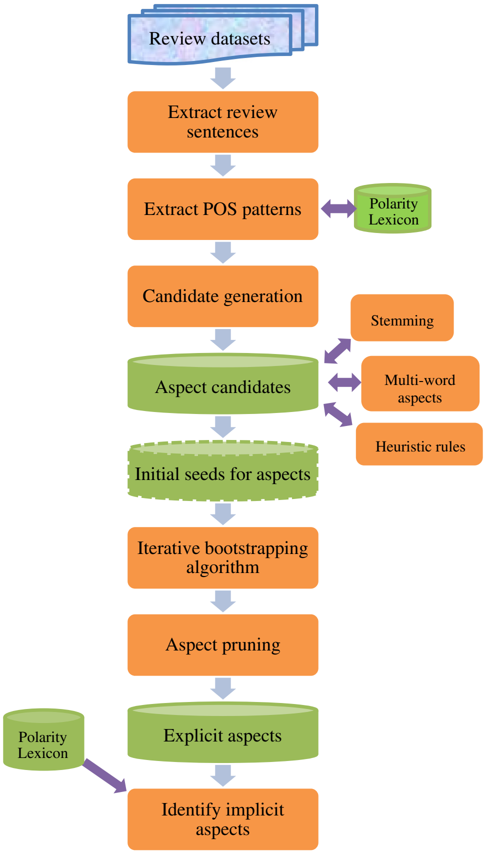


Fig. 3. The proposed model for aspect detection for sentiment analysis.

4.1. Part-of-Speech tagging and stemming

The model starts with extracting review sentences, and then for each of the sentences POS tagging is utilized, and candidates for aspects are extracted and stemmed. A Part-Of-Speech Tagger (POS Tagger) is a software package that reads text and assigns parts of speech tags to each word, such as noun, verb, adjective, etc. In this paper we focus on five POS tags: NN, JJ, DT, NNS and VBG, for nouns, adjectives, determiners, plural nouns and verb gerunds respectively [13]. Stemming is used to select one single form of a word instead of different forms. The goal of stemming is to reduce inflectional forms and sometimes derivationally related forms of a word to a common base form. In this work we use the Stanford software package for both POS tagging and stemming [26].

4.2. POS patterns and candidate generation

Based on the observation that aspects are nouns, in the model we extract combination of noun phrases and adjectives from review sentences. We use several experimentally extracted POS patterns which we introduce as heuristic combinations in Table 1.

From Table 1, heuristic combinations of the first row selects the candidate aspects from the noun phrase patterns like “NN”, “NNS”, “NN NN” and etc. The second row uses patterns like “JJ NN”, “JJ NNS”, “JJ NN NN” and so on. The third row of the Table 1 selects candidates based on the pattern “DT JJ”, and the last row of the table uses heuristic patterns like “DT VBG”, “VBG NN” and “NN VBG NN”.

Fig. 4 is a portion of an actual review for a digital camera from *www.amazon.com*. The phrases in bold are examples for the aspects we intend to extract. The phrase “digital camera” is extracted from the pattern “JJ NN”, phrases “picture quality” and “lens cover” are based on the pattern “NN NN”, and “zooming lever” extracted by “VBG NN” POS pattern.

4.3. Multi-word aspects

In the review sentences, some aspects that people talk about have more than one single word, “battery life”, “signal quality” and “battery charging system” are examples. This step is to find useful multi-word aspects from the reviews. A multi-word aspect is represented by $a = a_1, a_2, \ldots, a_n$ where a_i represents a single-word contained in a, and n is the number of words contained in a.

In this paper we propose a generalized version of FLR method [15] to rank the extracted multi-word aspects and select the importance ones. FLR is a word scoring method that uses internal structures and frequencies of candidates (FLR: Frequencies and Left and Right of the current word). One of the advantages of the FLR method is its size-robustness, that it can be applied to small corpus with less significant drop in performance than other standard methods like TF and IDF, because it is defined using more fine-grained features [30].

The FLR for an aspect a is calculated as:

$$FLR(a) = f(a) * LR(a) \qquad (1)$$

where $f(a)$ is the sentence frequency for aspect a, in other words it is the number of sentences that contain aspect a, and $LR(a)$ is the LR

Table 1
Heuristic combination POS patterns for candidate generation.

Description	Patterns
Nouns	Unigram to four-gram of NN and NNS
Nouns and adjectives	Bigram to four-gram of JJ, NN and NNS
Determiners and adjectives	Bigram of DT and JJ
Nouns and verb gerunds	Bigram to trigram of DT, NN, NNS and VBG

I just bought the ***digital camera*** *a few days ago. before I get used to it , here are my first feelings, the* ***picture quality*** *is so great, the* ***lens cover*** *is surely loose, the* ***zooming lever*** *is shaky , I hope it does not operate mechanically, otherwise you'll feel uneasy .*

Fig. 4. Sample digital camera review.

score of aspect *a* which is defined as a geometric mean of the scores of subset single-words as:

$$LR(a) = (lr(a_1) * lr(a_2) * \ldots * lr(a_n)) \wedge (1/n)$$

In this equation, each a_i represents a single-word in the multi-word aspect *a*, and *n* is the number of single-words in *a*. Note that LR is a measure for a multi-word aspect, whereas *lr* is a measure for subwords of the aspect.

An LR method is based on the intuition that some words are used as sub-words more frequently than others [30], and an aspect that contains such words is likely to be important. There are two versions for scoring with LR: Type-LR and Token-LR [30]. Type- and Token-LR can be calculated by counting the frequency of the types of words and frequency of the words connected to each word. In this word we apply Type-LR method for our proposed FLR. Therefore in the previous formula $lr(a_i)$ can be defined as:

$$lr(a_i) = \sqrt{l(a_i) * r(a_i)}$$

where the left score $l(a_i)$ of each word a_i of a target aspect is defined as the number of types of words appearing to the left of a_i, and the right score $r(a_i)$ is defined in the same manner.

As an example, consider the word "Sound" from Fig. 5, for which the Type-LR score is calculated as follows:

$$l(Sound) = 1, \quad r(Sound) = 4, \quad lr(Sound) = \sqrt{4}$$

And Token-LR score for "Sound" is:

$$l(Sound) = 2, \quad r(Sound) = 10, \quad lr(Sound) = \sqrt{20}$$

As the Type-LR can reflect the number of different types of words connected to the current word, here in our methods we use Type-LR score.

The main advantage of our proposed modified version of FLR is that it can extract all multi-word phrases and it is not only limited to two- or three-word phrases. The proposed modified and generalized FLR method is shown in Fig. 6. The generalization of this method is on the definition of two parameters: $l(a_i)$ and $r(a_i)$. We define $l(a_i)$ and $r(a_i)$ with regard to all words on the left and all word in the right of word *i* respectively. Therefore we change the definitions to give more importance to the aspects with more containing words. In the new definition, in addition to the frequency we consider position of a_i in aspect *a*. For the score $l(a_i)$ of each word a_i of a target aspect, we not only consider a single word to the left of a_i, but we check if there is more than one word on the left. We assign a weight for each position, that this weight is equal to one for the first word on the left, is two for the second word and so on. We define the score $r(a_i)$ in the same manner. In addition, we apply the add-one smoothing to both of them to avoid the score from being zero when a_i has no connected words.

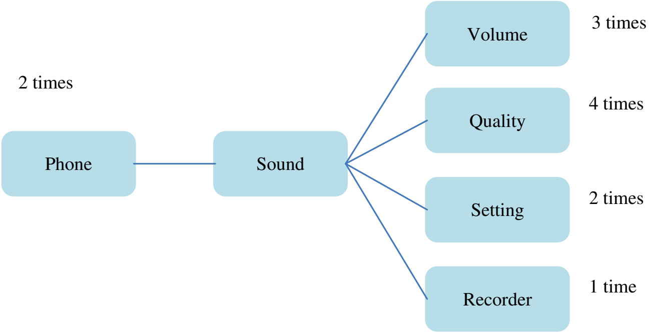


Fig. 5. An example for calculating Type-LR score.

```
Function modifiedFLRforAnAspect(aspect a)
BEGIN

define initialized FLR = 1;
define String s[] = aspect a split by space (" ");
if (aspect a contains only one word)      return FLR;
define three double arrays (l[], r[] and lr[]) each with the length equal to
the number of distinct words in aspect a;
for each distinct word i in aspect a
{
        l[i] = 1;
        r[i] = 1;
        for each aspect in the list of candidateAspects
        {
        define String x = candidateAspects[k];
        if(x contains s[i])
        define String xSplit = x split by space (" ");
        if(xSplit.length > 1)
        {
        search s[i] in x and set its index to j
        if (x contains more than two words)
        {
                if j shows the first word in x
                        r[i] += 1; // weight = 2
                else if j shows the last word in x
                        l[i] += 1;
                else if j shows a word in the middle of x
                {
                        l[i] += j; // weight = j
                        r[i] += j;
                }
        }
        else
        {
                if j shows the first word in x
                        r[i] += 1; // weight = 1
                else if j shows the last word in x
                        l[i] += 1;
                else if j shows a word in the middle of x
                {
                        l[i] += 1;
                        r[i] += 1;
                }
        }
        } // end if
        } // end for
         lr[i] = Math.sqrt(l[i]*r[i]);
        FLR = FLR*lr[i];
}
FLR = Frequency(aspect a)*Math.pow(FLR, 1/s.length);
return FLR;

END
```

Fig. 6. Modified version of FLR method for ranking multi-word aspects.

4.4. Heuristic rules

With finding the candidates, we need to move to the next level, aspect identification. For this matter we start with heuristic and experimentally extracted rules. Below, we discuss two rules in aspect detection model.

Rule #1: Remove aspects which there are no opinion words with in the sentence.
Rule #2: Remove aspects that contain stop words.

As the purpose of extracting aspects is to construct a sentiment analysis system, if no opinion words appear with the aspect phrase, the aspect is not very valuable. Therefore we employ Rule #1 for the proposed model. Opinion words are words that people use to present a positive or negative opinion. Most of the opinion words come as an *adjective* in sentence, hence in this study we check *adjective* phrases for opinion words in Rule #1, and therefore we extract *adjective* phrases from review sentences to construct a polarity lexicon. To illustrate the effect of Rule #1, we will demonstrate its working to the review sentences "signal strength will affect the battery life." and "battery life is very good, I use it every day and I have to charge it every 5 or 6 days or so." Both sentences talk about the aspect "battery life", the first sentence is not an opinionated sentence and tells a fact about battery life, whereas, the second sentence expresses an opinion or sentiment about "battery life". By applying Rule #1 we can ignore sentences without opinions like the first sentence for candidate aspect extraction.

With Rule #2 we remove candidate aspects that contain stop words as they are considered not to contribute any semantic weight. For instance, pattern "JJ NN" from Table 1 can extract some incorrect aspect candidates like "other phone". According to Rule #2 this "other phone" should be removed for the set of candidate aspects. In our experiment these heuristic rules turned out to improve the performance of aspect detection model.

4.5. Initial seeds for aspects

As mentioned above, our model is completely unsupervised and can do without any labeled data, but the bootstrapping algorithm needs some initial seeds for the input to find the rest of the aspects. Therefore we introduce A-score metric to extract a small list of aspects from the candidates as seed information. In our experiments we found that by using A-score, the top 10 highest values of the aspects could have perfect precision on the dataset. Hence, we focus on selecting some aspects from the candidates as seed set information by using an unsupervised metric, the A-score. The initial seed set is the input for the iterative bootstrapping algorithm in the model.

4.6. A-score metric

In this paper we introduce a new approach for evaluating aspect candidates, the A-score. In order to explain the A-score measure, it is helpful to first introduce PMI.

4.6.1. Pointwise Mutual Information (PMI)

PMI is a measure of association used in information theory and statistics. The PMI value of two random variables is a quantity that measures the mutual dependence of the two random variables. In other words, for the field of text mining PMI compares the probability of observing two words together with the probabilities of observing two words independently. So it can be used to estimate whether the two words have a genuine association or just be observed by chance [22]. Researchers have applied this measurement to many natural language processing and data mining problems such as feature selection and text classification.

PMI can be computed by:

$$PMI(a,b) = \log\frac{P(a,b)}{P(a)P(b)} \tag{2}$$

where $P(a, b)$ is the probability of co-occurrence of word a and word b together, and $P(a)$ and $P(b)$ are the probability of occurrence of word a and word b in the review documents respectively. Usually, word probabilities $P(a)$, $P(b)$ and joint probabilities $P(a, b)$ can be estimated by counting the number of observations of a, b and the co-occurrence of a and b in a corpus normalizing by the size of the corpus.

Intuitively PMI measures if the co-occurrence of a and b is more likely than their independent occurrences. The quality of the PMI algorithm largely depends on the size of training data. If there is no co-occurrence of a and b words in the corpus, the accuracy of PMI becomes an issue.

4.6.2. A-score

Here we introduce a new metric, named A-score which is originally based on PMI, but A-score uses both frequency-based and inter-relation information between words to score them. We score each candidate aspect with A-score metric defined as:

$$\mathrm{A-Score}(a) = f(a) * \Sigma_i \log_2([f(a,b_i)/f(a) * f(b_i)] * N + 1) \tag{3}$$

where a is the current aspect, $f(a)$ is the number of the sentences in the corpus which a is appeared, $f(a, b_i)$ is the frequency of co-occurrence of aspect a and b_i in each sentence. b_i is ith aspect in the list of Seed Aspects, and N is number of sentences in the corpus. The A-score metric is based on mutual information between an aspect and a list of aspects, in addition it considers the frequency of each aspect. We apply the add-one smoothing to the metric, so all co-frequencies are non-zero. This metric helps to extract more informative aspects and more co-related ones. We use A-score metric to extract both initial Seed Aspects and Final Aspects. When we utilize A-score to extract initial Seed Aspects b_i is ith aspect in the list of Candidate Aspects.

4.7. Iterative bootstrapping algorithm for detecting aspects

The iterative bootstrapping algorithm focuses on learning the ultimate list of aspects from a small amount of unsupervised seed data. Bootstrapping can be viewed as an iterative clustering technique for which in each iteration, the most interesting and valuable candidate is chosen to adjust the current seed set. This technique continues until satisfying a stopping criterion like a predefined number of outputs. A crucial task for an iterative bootstrapping algorithm is how to measure the value score of each candidate in each iteration. The proposed iterative bootstrapping algorithm for detecting aspects is shown in Fig. 7. In this algorithm we use A-score metric to measure the value score of each candidate in each iteration.

From Fig. 7, the task of the proposed iterative bootstrapping algorithm is to enlarge the initial seed set and generate a final list of aspects. In each of the iterations, the current version of the seed set and the list of candidate aspects are used to find the value score of A-Score metric for each candidate, resulting one more aspect for the seed set. Finally, the augmented seed set is the final aspect list and the output of the algorithm.

4.8. Aspect pruning

After finalizing the list of aspects, there may exist redundant selected ones. For instances, "Suite" or "Free Speakerphone" are both redundant aspects, while "PC Suite[1]" and "Speakerphone" are meaningful ones. Aspect pruning aims to remove these kinds of redundant aspects. For aspect pruning, we introduce two kinds of pruning below.

4.8.1. Subset-Support pruning

As we can see from Table 1, two of the POS patterns are "JJ NN" and "JJ NN NN". These patterns extract some useful and important aspects like "remote control" or "optical zoom", but there are some redundant and meaningless aspects regarding to these patterns. Aspects like "free speakerphone" or "rental dvd player" are

[1] PC Suite is a software package used to establish an interface between mobile devices like a cellphone and computers.

```
Algorithm: A-Score Iterative Bootstrapping for Detecting Aspects
Input: Seed Aspects, Candidate Aspects
Output: Final Aspects
Method:
        While(stopping criterion)
            For each aspect in Candidate Aspects
                A = A-score(aspect)
                If A > max then max = A
            End For
            Add aspect with maximum A-score to the Seed Aspects
        End While
        Copy Seed Aspects to the list of Final Aspects
Return Final Aspects
```

Fig. 7. The proposed iterative bootstrapping algorithm for detecting aspects.

examples, while subsets of them “speakerphone” or “dvd player” are useful aspects. This step checks multi-word aspects that start with an adjective (JJ POS pattern), and removes those that are likely to be meaningless. In this step we remove the adjective part for aspects and then check a threshold if the second part is meaningful, we introduce a threshold named Subset-Support for the aspect a as the following,

$$\text{Subset} - \text{Support}(a) = (\text{count of the aspect a} + 1)/(\text{count of remaining part of aspect a} + 1) \tag{4}$$

Experiments show using this step will increase accuracy of the model.

4.8.2. Superset-support pruning

In this step we remove redundant single word aspects. We filter single word aspects which there are superset ones of them. “Suite” or “life” are both examples of these redundant aspects which “PC Suite” or “battery life” are superset meaningful ones. Superset-Support pruning is like redundancy pruning in [7], but the difference is we remove those rare single word aspects which their frequency ratio to the frequency of the superset is below than an experimentally threshold set one. The experiments show removing based on the ratio on superset aspect will give us more accurate list of aspects.

4.9. Identification of implicit aspects

This section is focused on identifying implicit aspects in reviews. In this paper, like Su et al.’s work (2008), we consider that an implicit aspect should satisfy the following conditions:

- The related aspect word does not occur in the review sentence explicitly.
- The aspect can be discovered by its surrounding words (e.g. opinion words) in the review sentence.

Table 2 shows three examples of implicit aspects in review sentences for Nokia 6610 from *www.amazon.com*.

We propose a graph-based approach for identifying implicit aspects in the reviews. By utilizing a polarity lexicon and a list of predefined aspects, we draw a graph for aspects and opinion words (see Fig. 8). The graph uses an opinion word from the polarity lexicon as a node and maps this node to the set of the aspects nodes. In the graph, we set an edge to a pair of nodes if they co-occur together in a review sentence, and we assign initial weight *w* to the edge as the number of their co-occurrence. In the proposed approach we use extracted aspects and opinion words from the previous sections.

Table 2
Examples of implicit aspects in review sentences for Nokia 6610.

Review sentence	Implicit aspect
It is small	Size
I like my phone to be small so I can fit it in my pockets	Size
This is a very light phone	Weight

Using only the co-occurrence of aspect and opinion word for identifying implicit aspects are not enough, therefore we define a function to measure the association of an aspect and opinion word as:

$$co - occurrence(aspect, opinion\ word) = \log\left(\frac{w_{aspect,opinion\ word}}{degree_{aspect} * degree_{opinion\ word}} + \varepsilon\right) \tag{5}$$

In this approach we use this function to update the weights in the graph. In the function, $w_{aspect,\ opinion\ word}$ is the current weight of the edge between node *aspect* and node *opinion word*, $degree_{aspect}$ is the number of distinct opinion words that appear with the aspect, $degree_{opinion\ word}$ is the number of distinct aspects that appear with the opinion word and ϵ is a parameter to prevent the fraction from getting zero when the $w_{aspect,\ opinion\ word}$ is zero. The *co-occurrence* measure compares the probability of observing aspect and opinion word together with the probabilities of observing them independently. We use value of the *co-occurrence* measure to update the weight of the edge between node *aspect* and node *opinion word*.

In the approach, after updating the weights of edges in the graph, we define a gap-threshold to describe the margin between co-occurrence measures values of aspects for an opinion word. Based on the gap-threshold we extract a list of most likely implicit aspects for each of the opinion words in the polarity lexicon.

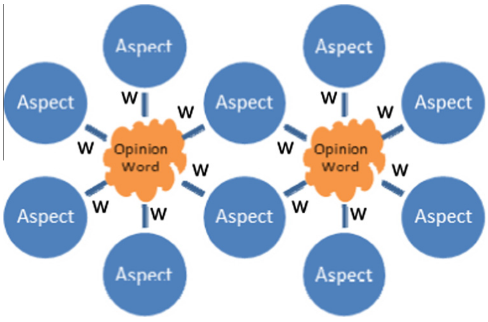


Fig. 8. A sample of association graph between aspects and opinion words.

5. Experimental results

In this section we discuss the experimental results for the proposed model and presented algorithms. To report the effectiveness of our model first we evaluate the results for each individual step in of our model, and then we compare the results with the benchmarked results by Wei et al. [27] and Somprasertsri and Lalitrojwong's [21]. Finally we discuss about identification of implicit aspects. In the following, data collection, evaluation measures and important evaluation results will be discussed.

5.1. Data collection

We employed datasets of customer reviews for five products with feature annotations for our evaluation purpose [7,17]. This dataset focus on different domain of electronic products: Apex AD2600 Progressive-scan DVD player, Canon G3, Creative Labs Nomad Jukebox Zen Xtra 40 GB, Nikon Coolpix 4300, and Nokia 6610, and have been widely used by researchers for opinion mining. Table 3 shows number of reviews, number of review sentences and the number of manually tagged product aspects for each product in this dataset. Since these five datasets are small for aspect detection in review mining, we crawled many other product reviews from *Amazon.com* and *cnet.com*. The details of each dataset are given in Table 4. Newly extracted product reviews are from the same domain as Table 3, but the difference is that they are not from the same specific product but from similar series of the product. For example 'Canon' in Table 4 shows reviews for 'Canon g3', 'Canon g6', 'Canon g7' and 'Canon g10'.

Since the product features in the customer review datasets in Table 3 have already been annotated by human annotators, these annotated product features form a gold standard for the corresponding domain and we use them as reference values for each dataset.

5.2. Evaluation measures

We use precision, recall, F-score and accuracy to measure the effectiveness of an aspect extraction technique.

The precision, recall and F-score are calculated based on Table 5 as:

$$Precision = TP/(TP + FP) \tag{6}$$

$$Recall = TP/(TP + FN) \tag{7}$$

$$F - score = (2 * (Precision * Recall))/(Precision + Recall) \tag{8}$$

The accuracy is the proportion of true results (both true positives and true negatives) in the population. It is computed based on Table 5 as:

$$Accuracy = (TP + TN)/(TN + FP + FN + TP) \tag{9}$$

To deal with multiple datasets (products), we adopt the macro and micro average [28] to assess the overall performance. The macro- and micro-averaged precision and recall across the *n* datasets are defined as follows:

$$Macro - averaged\ precision = \frac{\sum_{i=1}^{n} Precision_i}{n} \tag{10}$$

$$Macro - averaged\ recall = \frac{\sum_{i=1}^{n} Recall_i}{n} \tag{11}$$

$$Micro - averaged\ precision = \frac{\sum_{i=1}^{n} TP_i}{\sum_{i=1}^{n} (TP_i + FP_i)} \tag{12}$$

$$Micro - averaged\ recall = \frac{\sum_{i=1}^{n} TP_i}{\sum_{i=1}^{n} (TP_i + FN_i)} \tag{13}$$

Table 3
Summary of customer review datasets.

Dataset	Number of reviews	Number of review sentences	Number of manual aspects
Canon G3	45	597	100
Nikon Coolpix 4300	34	346	74
Nokia 6610	41	546	109
Creative Labs Nomad Jukebox Zen Xtra	95	1716	180
Apex AD2600	99	740	110

Table 4
Detailed information of the five review datasets.

Dataset	Number of reviews	Number of review sentences
Canon	484	5875
Nikon	503	4335
Nokia	471	4069
Creative	525	6206
Apex	497	4124

The macro average is calculated by simply taking the average obtained for each dataset, which gives an equal weight for every dataset and product. Whereas the micro average assigns each dataset a relative weight on the basis of the number of extracted or manually tagged aspects for the dataset.

5.3. Comparative study

In our evaluation, after preprocessing and extracting the candidates, we score each multi-word aspect with the generalized FLR method and select those with the score higher than the average, and then we merge single-word and multi-word aspects in a list. Heuristic rules are then employed for the whole list of single- and multi-word aspects to take into account the influence of an opinion word on detecting the aspect and remove useless aspects.

Finding an appropriate number of good seeds for bootstrapping algorithm is an important step. In our experiments we used A-score metric to extract automatically the seed set. We have experimented with different numbers of seeds (i.e., 5, 10, 15 and 20) for iterative bootstrapping, and found that the best number of the seeds is about 10–15. Therefore seeds were automatically chosen for iterative bootstrapping algorithm, and the stopping criterion is defined when about 70–120 aspects have been learned. For the subset-support pruning method we set the threshold 0.5. In superset-support pruning step if an aspect has a frequency lower than three and its ratio to the superset aspect is less than experimentally threshold set one, it is pruned. Table 6 shows the experimental results of our model at three main steps described in Section 3, Multi-word aspects and heuristic rules, Iterative bootstrapping with A-score and Aspect pruning steps.

Table 6 gives all the precision and recall results at the main steps of the proposed model for explicit aspect extraction. In this table, column 1 lists each product. Each column gives the precision and recall for each product. Column 2 uses extracted single-word aspects and selected multi-word aspects based on generalized FLR approach and employing heuristic rules for each product. The results indicate that extracted aspects contain a lot of errors.

Table 5
Contingency table.

	Not predicted as aspect	Predicted as aspect
Wrong aspects	TN	FP
Correct aspects	FN	TP

Using this step alone gives poor results in precision. Column 3 shows the corresponding results after employing Iterative bootstrapping algorithm with A-score metric. We can see that the precision is improved significantly by this step but the recall drops. Column 4 gives the results after pruning methods are performed. The results demonstrate the effectiveness of the pruning methods. The precision is improved dramatically, but the recall drops a few percent.

5.3.1. Comparing the model to Wei et al.'s work

We evaluated the effectiveness of the proposed model and compared the results with the benchmarked results by Wei et al. [27]. Wei et al. proposed a semantic-based product aspect extraction (SPE) method and compared the results of the SPE with the association rule mining approach (ARM) given in [7]. The SPE technique exploits a list of positive and negative adjectives defined in the General Inquirer to recognize opinion words semantically and subsequently extract product aspects expressed in customer reviews.

Table 7 shows the experimental results of our model on precision (P) and recall (R) in comparison with SPE and ARM techniques for the customer review datasets. Both the ARM and SPE techniques employ a minimum support threshold set at 1% in the frequent aspect identification step for finding aspects according to the association rule mining. From this table we can see that the proposed unsupervised model outperforms other techniques in precision, but in recall SPE has a slightly better performance. To assess the overall performance of techniques, Figs. 9 and 10 show the macro and micro average for precision and recall respectively.

From the figures, the macro-averaged precision and recall of the existing ARM technique are 47.9% and 60.9% respectively, whereas the macro-averaged for precision and recall of the SPE technique are 49.8% and 71.6% respectively. Thus the effectiveness of SPE is better than that of the ARM technique, recording improvements in macro-averaged precision and recall. However, our proposed model outperforms both benchmark techniques in precision, achieving a macro-averaged precision of 84.5%. Specifically, macro-averaged precision obtained by the proposed model is 36.6% and 34.7% higher than those reached by the existing ARM technique and SPE, respectively. The proposed model reaches to a macro-averaged recall at 66.5%, where improves the ARM by 5.6%, but it is about 5.1% less than SPE approach. When considering the micro average measures, we observe similar results to those we obtained by using macro average measures.

It is notable that the proposed model and techniques have a more substantial improvement in precision than in recall. In the other words, our model makes significant improvements over other techniques in all the datasets in precision, but in recall SPE has better performance. For example, our model records 36.6% and 34.7% improvements in terms of macro-averaged precision over the ARM and SPE techniques respectively, and 38.1% and 35.6% improvements in terms of micro-averaged precision. However, the proposed model achieves an averagely higher recall than the ARM technique but a slightly lower recall than the SPE technique. One reason is that for the iterative bootstrapping algorithm we limit the number of output aspects between 70 and 120 aspects, therefore the precision for the output will be better than the recall.

Additionally, Fig. 11 shows the F-score values of different approaches using different product datasets. In all five datasets, our model achieves the highest F-score. This indicates that our unsupervised model is effective in extracting aspects and is superior to the existing techniques. We can thus draw the conclusion that our model can be used in practical settings, in particular those where high precision is required.

Table 6
Recall and precision for three main steps of the proposed model.

Dataset	Multi-word aspects and heuristic rules	Iterative bootstrapping with A-score	Aspect pruning
Precision			
Canon	26.7	75.5	84.1
Nikon	28.4	69.8	87.5
Nokia	23.9	73.5	79.0
Creative	14.8	80.1	90.0
Apex	19.3	78.8	82.0
Average	22.62	75.54	84.52
Recall			
Canon	85.7	74.0	70.7
Nikon	82.4	72.5	68.6
Nokia	84.1	72.5	71.0
Creative	78.9	59.2	57.1
Apex	74.6	65.1	65.1
Average	81.14	68.66	66.5

This comparative evaluation suggests that the proposed model, which involves frequency-based and inter-relation information between the aspects and gives more importance to multi-word aspects and uses the influence of an opinion word in the review sentence, attains better effectiveness for product aspect extraction. The existing ARM technique depends on the frequencies of nouns or noun phrases for the aspect extraction, and SPE relies primarily on frequency- and semantic-based extraction of noun phrases for the aspect detection. For Example, our model is effective in detecting aspects such as ‘‘digital camera’’ or ‘‘battery charging system’’, which both ARM and SPE failed on extraction of these aspect phrases. Additionally, we can tune the parameters in our model to extract multi-word aspects with less or more words. For example aspect ‘‘canon power shot g3’’ can be found by the model. Therefore, the results show that using a completely unsupervised approach for aspect detection in sentiment analysis could achieve promising performances.

5.3.2. Comparing the model to a supervised technique

As mentioned before, the proposed model is an unsupervised domain-independent model. We therefore empirically investigate the performance of using a supervised technique for aspect detection in comparison to the proposed model. We employ results of a supervised technique from Somprasertsri and Lalitrojwong's work (2008). They proposed an approach for aspect detection by combining lexical and syntactic features with a maximum entropy model. Their approach uses the same dataset collection of product reviews we experimented on. They extract the learning features from the annotated corpus of Canon G3 and Creative Labs Nomad Jukebox Zen Xtra 40 GB from customer review dataset. In their work, the set of data was split into a training set of 80% and a testing set of 20%. They employed the Maxent version 2.4.0 as the classification tool. Table 8 shows the micro-averaged precision, micro-averaged recall and micro-averaged F-score of their system output in comparison to our proposed model for the Canon and Creative datasets.

Table 7
Experiment results of comparative study.

Product	ARM		SPE		Proposed model	
	P	R	P	R	P	R
Canon	51.1	63.0	48.7	75.0	84.1	70.7
Nikon	51.0	67.6	47.4	75.7	87.5	68.6
Nokia	49.5	57.8	56.5	72.5	79.0	71.0
Creative	37.0	56.1	44.0	65.0	90.0	57.1
Apex	51.0	60.0	52.4	70.0	82.0	65.1

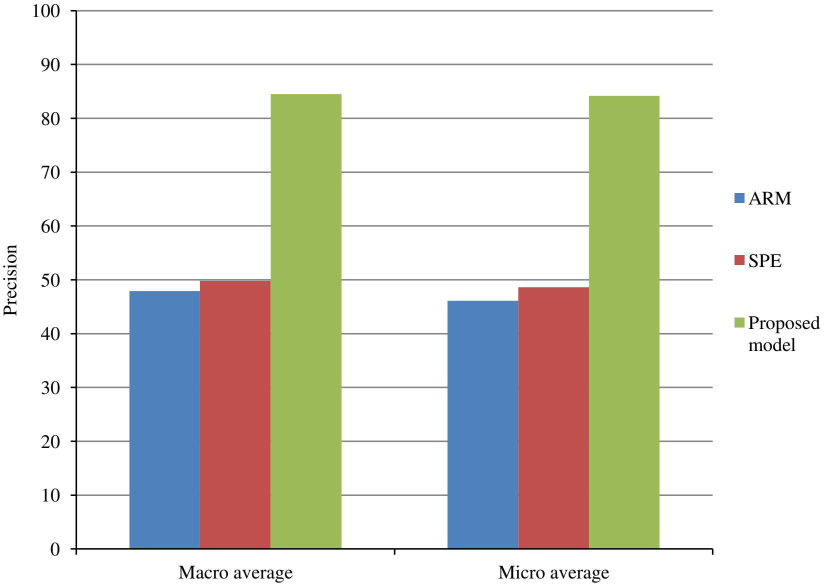


Fig. 9. Macro and Micro average for precision.

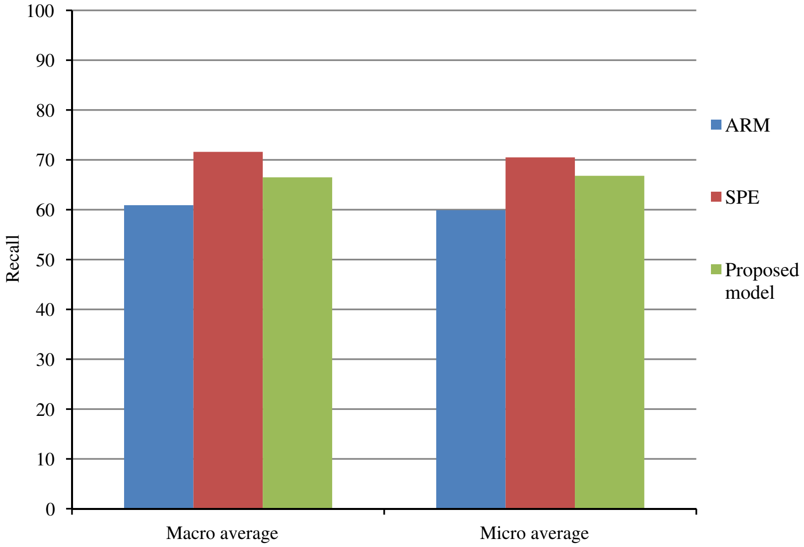


Fig. 10. Macro and Micro average for recall.

Table 8 shows that for the proposed model, the precision is improved dramatically by 15.9%, the recall is decreased by 4.1% and the F-score is increased by 4.3%. Therefore our proposed model and presented algorithms outperforms the Somprasertsri and Lalitrojwong's model. The significant difference between our model and theirs is that they use a fully supervised structure for aspect detection, while our proposed model is completely unsupervised and domain-independent. In most applications the supervised techniques can achieve reasonable effectiveness, but preparing training dataset is a time consuming task and the effectiveness of the supervised techniques greatly depends on the representativeness and labeling of the training data. In contrast, unsupervised models automatically extract product aspects from customer reviews without needing any training data. Moreover, the unsupervised models seem to be more flexible and wider applicable than the supervised ones, in particular for environments in which various and frequently expanding products or services get discussed in customer reviews.

5.4. Analyzing implicit aspect identification

We also evaluated the identification of implicit aspects for the customer review dataset. Here for the discussion and analysis of the implicit aspect identification method we focus more on the review dataset of Nokia 6610 product. First we extract a list of co-occurred opinion words with aspects for each extracted explicit aspect from the previous section as the polarity lexicon and then we create an association graph of aspects and opinion words based on the co-occurrence frequency of them as initial weights. Table 9 shows an example of the initial weights for the five selected opinion words and six selected aspects in the graph for Nokia 6610 product.

Based on the initial weights from Table 9 and the co-occurrence metric we calculate the new weights for the graph and select a list of most likely implicit aspects for each opinion word. Table 10 shows co-occurrence metric values for the examples in Table 9.

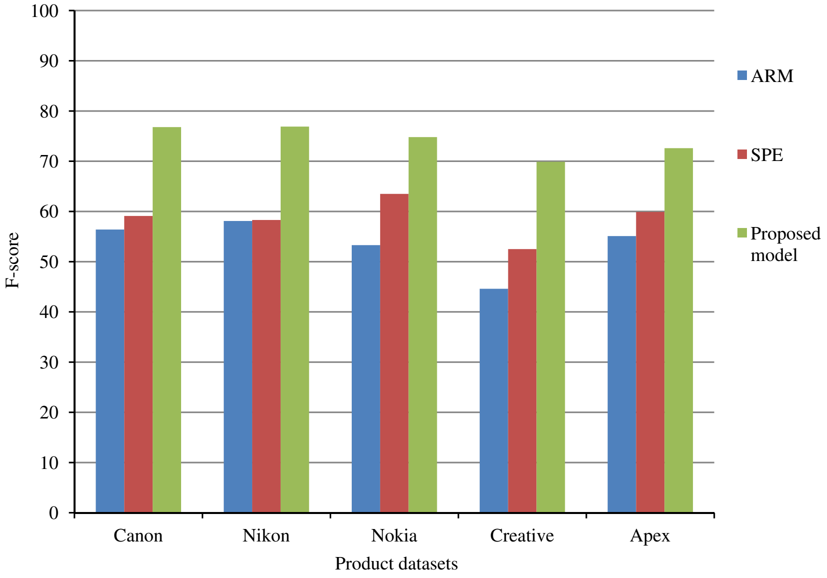


Fig. 11. F-scores of ARM, SPE, and the proposed model for each dataset.

Table 8
Micro-averaged precision, recall and F-score for supervised maximum entropy and our unsupervised model.

	Precision	Recall	F-score
Maximum entropy model	71.6	69.1	70.3
Proposed model	87.5	65	74.6

For each opinion word, we have got its updated weight on each aspect. We need to further map them to one or several most suitable aspects according to their co-occurrence scores and gap-threshold. In our experiments, we set the gap-threshold to 0.01. When the margin is greater than the gap-threshold we say that the two aspects have a gap. With each opinion word and co-occurrence scores of them with aspects, we find every gap value from high to low, and judge the opinion word which the aspect should belong to according to the sequence. We use PMI to compare the results of our approach to for five selected opinion words and six selected aspects. The difference between our approach and the PMI measure is that we use distinct occurrences of words and distinct co-occurrences of them, but the PMI uses the number of all observations. In addition, we use a parameter ϵ to prevent the fraction from getting zero. Table 11 shows PMI scores for the examples in Table 9.

Based on the proposed approach, we map the opinion words in our polarity lexicon to the pre-extracted list of aspects. For comparison we use PMI measure for the customer product review dataset. In the dataset there are 156 opinionated review sentences that the aspect not appeared in the sentence, which our proposed approach reaches to 74.36% accuracy for detecting implicit aspects and the PMI measures to 67.95% accuracy. In our experiments, the proposed approach for identifying implicit aspects reaches to more accurate lists of aspects than the PMI measure. Table 12 gives some experimental example results of sorted candidate implicit aspects for opinion words based on the co-occurrence and PMI scores for the Nokia 6610 review dataset.

5.5. Analyzing aspects in ironic reviews

The final analysis of the proposed model is to study explicit and implicit aspects from ironic reviews. Ironic review is a sophisticated form of opinionated text in which the author conveys his/her opinion in an implicit way. Two examples for the ironic reviews are:

1. "It's not that there is not anything positive to say about the film. There is. After 92 min, it ends!"
2. "Everything else about the camera is great!"

These two examples, according to some user-generated labels, could be ironic or sarcastic. However, the issue we want to focus on does not lie in detecting ironic expressions in every review, but rather in finding what aspects most of the reviews are about, i.e. the first review is talking about the aspect "film" and the second one is about aspect "camera".

We extracted manually a subset of ironic reviews from the dataset in Table 3 to apply the proposed model to detect explicit and implicit aspects. We scanned the collection and filter out the reviews with rating less than four stars. The reasons for this decision rely on the viral purpose and the ironic effect. The viral purpose causes that people to post reviews which the main purpose is to raise superficial properties and non-existent effects. The ironic effect assumes that if someone ironically wants to reflect properties or features of an object he or she does not do it by rating the products with one or two stars. Instead he or she rates them with the highest scores [19]. After applying this filter, an ironic set of reviews extracted manually, which their frequencies are much less than 10% of total reviews.

The identification of explicit and implicit aspects for the ironic review set is evaluated to find out what explicit or implicit aspects most of the ironic reviews are about. Table 13 gives some experimental example results of aspects for the ironic set of reviews.

Table 9
Initial weights for some opinion word and aspect examples in the association graph.

	Battery life	Sound quality	Size	Phone	Price	Signal	Weight
Great	2	2	0	16	1	1	0
Good	3	4	1	6	0	1	0
Small	0	0	4	6	0	0	0
Excellent	0	2	0	2	0	1	0
Perfect	0	0	3	4	0	1	0
Light	0	0	0	4	0	0	4

Table 10
Co-occurrence metric values (updated weights) for the examples in Table 8.

	Battery life	Sound quality	Size	Phone	Price	Signal	Weight
Great	0.051530	0.036994	0	0.062228	0.064130	0.025995	0
Good	0.060542	0.057715	0.020464	0.018616	0	0.020464	0
Small	0	0	0.341037	0.084889	0	0	0
Excellent	0	0.045088	0	0.009683	0	0.031709	0
Perfect	0	0	0.137504	0.028855	0	0.047306	0
Light	0	0	0	0.043069	0	0	0.584963

Table 11
PMI scores for the examples in Table 8.

	Battery life	Sound quality	Size	Phone	Price	Signal	Weight
Great	0.010647	0.008878	0	0.006227	0.026472	0.010647	0
Good	0.018696	0.020759	0.005691	0.002745	0	0.012491	0
Small	0	0	0.028855	0.003506	0	0	0
Excellent	0	0.018379	0	0.001619	0	0.022026	0
Perfect	0	0	0.029953	0.003237	0	0.022026	0
Light	0	0	0	0.003824	0	0	0.101283

Table 12
Mapping results of some example opinion words.

Opinion word	List of aspects based on PMI	List of aspects based on proposed approach
Great	Price	Price, phone
Good	Sound quality, battery life, signal, size, phone	Battery life, sound quality
Small	Size	Size
Excellent	Signal, sound quality	Sound quality, signal
Perfect	Size, signal	Size
Light	Weight	Weight

Table 13
Some experimental example results of aspects for the ironic reviews.

Wireless telephone	Menu option
Reception	User interface
Phone	Screen
Customer service	Picture
Key	Player

By analyzing the results we established that the aspects discussed in ironic reviews are the same as standard reviews. For example, the proposed model extracted aspect “phone” from the ironic review sentence “I can routinely talk on the phone in my house, which is an unaccustomed luxury!” The main differences between an ironic and standard text are about the way authors express their sentiment, use opinion words or negations. In other words, irony is a playful use of language in which a speaker implies the opposite of what is literally said about same aspects as in standard reviews, i.e. a type of indirect negation or a sentiment in direct opposition to what is actually believed. The main problem for ironic reviews is about assigning either positive or negative polarity to determine what the truth value of a certain statement is.

6. Conclusions

In this research we study sentiment analysis and opinion mining for online reviews. When dealing with mining online reviews, it is often expensive and time consuming to construct labeled data for training purposes and it is desirable to develop a model or algorithm that can do without labeled data. In this paper we therefore proposed an unsupervised domain- and language-independent model for detecting explicit and implicit aspects from the reviews. The proposed model is able to deal with three major bottlenecks: domain dependency, the need for labeled data, and implicit aspects. We proposed a number of novel techniques for mining aspects from reviews. We used the inter-relation information between words in a review and the influence of an opinion word on detecting an explicit aspect. Furthermore we described an approach which uses a co-occurrence metric to calculate the association between opinion words and explicit aspect to identify implicit aspects. Our experimental results indicate that our model is effective in performing the task and outperforms other techniques. We can draw the conclusion that the model can be used in practical settings, in particular those where high precision is required.

In our future work, we plan to further improve and refine our model for aspect detection, opinion word detection and opinion word orientation identification. We have also planned a study on ironic review mining to investigate about explicit/implicit aspects of ironic reviews of a product review dataset. We plan to employ clustering methods in conjunction with the proposed model in this paper to extract the explicit and implicit aspects and explicit and implicit opinion words together to summarize output based on the opinions that have been expressed in the reviews. Finally, we will try to build an opinion summarization system which could works on different domains of different languages like English, Persian or Dutch.

Acknowledgments

The authors thank Dr. Djoerd Hiemstra for his invaluable comments and suggestions and gratefully acknowledge the hospitality offered to the first author by the Human Media Interaction (HMI) group at the University of Twente. The research of the last author of this paper is partially supported by the Dutch National FES Program COMMIT.

References

[1] A. Balahur, J.M. Hermida, A. Montoyo, Detecting implicit expressions of emotion in text: a comparative analysis, Decision Support Systems 53 (2012) 742–753.

[2] C. Bosco, V. Patti, A. Bolioli, Developing corpora for sentiment analysis and opinion mining: the case of irony and Senti-TUT, IEEE Intelligent Systems 28 (2) (2013) 55–63.

[3] S. Brody, N. Elhadad, An unsupervised aspect-sentiment model for online reviews, in: Proceedings of Annual Conference of the North American Chapter of the Association for Computational Linguistics, Publishing, Association for Computational Linguistics, 2010, pp. 804–812.

[4] T. Dunning, Accurate methods for the statistics of surprise and coincidence, Computational Linguistics 19 (1) (1993) 61–74.

[5] X. Fu, G. Liu, Y. Guo, Z. Wang, Multi-aspect sentiment analysis for Chinese online social reviews based on topic modeling and HowNet lexicon, Knowledge-Based Systems 37 (2013) 186–195.

[6] A. Hogenboom, F. Boon, F. Frasincar, A statistical approach to star rating classification of sentiment, Management Intelligent Systems (2012) 251–260.

[7] M. Hu, B. Liu, Mining opinion features in customer reviews, in: Proceedings of 19th National Conference on Artificial Intelligence, Publishing, AAAI Press, 2004, pp. 755–760.

[8] M. Hu, B. Liu, Mining and summarizing customer reviews, in: Proceedings of International Conference on Knowledge Discovery and Data Mining, Publishing, ACM SIGKDD, 2004, pp. 168–177.

[9] B. Liu, M. Hu, J. Cheng, Opinion observer: analyzing and comparing opinions on the web, in: Proceedings of International Conference of World Wide Web, Publishing, 2005, pp. 342–351.

[10] B. Liu, L. Zhang, A survey of opinion mining and sentiment analysis, Mining Text Data (2012) 415–463.

[11] C. Lin, Y. He, R. Everson, S. Ruger, Weakly supervised joint sentiment-topic detection from text, IEEE Transactions on Knowledge and Data Engineering 24 (6) (2012) 1134–1145.

[12] G. Mangnoesing, A. van Bunningen, A. Hogenboom, F. Hogenboom, F. Frasincar, An empirical study for determining relevant features for sentiment summarization of online conversational documents, in: Proceedings of 13th International Conference on Web Information Systems Engineering, 2012, pp. 567–579.

[13] M.P. Marcus, M.A. Marcinkiewicz, B. Santorini, Building a large annotated corpus of English: the Penn Treebank, Computational Linguistics 19 (2) (1993) 313–330.

[14] S. Moghaddam, M. Ester, ILDA: interdependent LDA model for learning latent aspects and their ratings from online product reviews, in: Proceedings of the 34th International ACM SIGIR Conference on Research and Development in Information Retrieval, Publishing, ACM, 2011, pp. 665–674.

[15] H. Nakagawa, T. Mori, Automatic term recognition based on statistics of compound nouns and their components, Terminology 9 (2) (2003) 201–219.

[16] A.M. Popescu, O. Etzioni, Extracting product features and opinions from reviews, in: Proceedings of Conference on Empirical Methods in Natural Language Processing, Publishing, Association for Computational Linguistics, 2005, pp. 339–346.

[17] G. Qiu, B. Liu, J. Bu, C. Chen, Opinion word expansion and target extraction through double propagation, Computational Linguistics 37 (1) (2011) 9–27.

[18] A. Reyes, P. Rosso, D. Buscaldi, From humor recognition to irony detection: the figurative language of social media, Data & Knowledge Engineering 74 (2012) 1–12.

[19] A. Reyes, P. Rosso, Making objective decisions from subjective data: detecting irony in customers reviews, Decision Support Systems 53 (2012) 754–760.

[20] E. Riloff, R. Jones, Learning dictionaries for information extraction by multi-level bootstrapping, in: Proceedings of 16th National Conference on Artificial Intelligence, Publishing, John Wiley & Sons LTD, 1999, pp. 474–479.

[21] G. Somprasertsri, P. Lalitrojwong, Automatic product feature extraction from online product reviews using maximum entropy with lexical and syntactic features, in: Proceedings of IEEE International Conference on Information Reuse and Integration, Publishing, 2008, pp. 250–255.

[22] Q. Su, K. Xiang, H. Wang, B. Sun, S. Yu, Using pointwise mutual information to identify implicit features in customer reviews, in: Proceedings of 17th International Conference on Computer Processing of Oriental Languages, Publishing, 2006, pp. 22–30.

[23] Q. Su, X. Xu, H. Guo, Z. Guo, X. Wu, X. Zhang, Z. Su, Hidden sentiment association in chinese web opinion mining, in: Proceedings of 17th International Conference on World Wide Web, Publishing, Beijing, China, 2008, pp. 959–968.

[24] T.T. Thet, J.C. Na, C.S. Khoo, Aspect-based sentiment analysis of movie reviews on discussion boards, Journal of Information Science 36 (6) (2010) 823–848.

[25] I. Titov, R. McDonald, A joint model of text and aspect ratings for sentiment summarization, in: Proceedings of the Annual Meeting on Association for Computational Linguistics and the Human Language Technology Conference (ACL-HLT), Publishing, 2008, pp. 308–316.

[26] K. Toutanova, D. Klein, C.D. Manning, Y. Singer, Feature-rich part-of-speech tagging with a cyclic dependency network, in: Proceedings of HLT-NAACL, Publishing, 2003, pp. 252–259.

[27] C.P. Wei, Y.M. Chen, C.S. Yang, C.C. Yang, Understanding what concerns consumers: a semantic approach to product feature extraction from consumer reviews, Information Systems and E-Business Management 8 (2) (2010) 149–167.

[28] Y. Yang, An evaluation of statistical approaches to text categorization, Information Retrieval 1 (1) (1999) 69–90.

[29] J. Yi, T. Nasukawa, R. Bunescu, W. Niblack, Sentiment analyzer: extracting sentiments about a given topic using natural language processing techniques, in: Proceedings of Third IEEE International Conference on Data Mining, Publishing, 2003, pp. 427–434.

[30] M. Yoshida, H. Nakagawa, Automatic term extraction based on perplexity of compound words, Natural Language Processing (2005) 269–279.

[31] J. Yu, Z.J. Zha, M. Wang, T.S. Chua, Aspect ranking: identifying important product aspects from online consumer reviews, Computational Linguistics (2011) 1496–1505.

[32] C. Zhai, J. Lafferty, Model-based feedback in the language modeling approach to information retrieval, in: Proceedings of 10th International Conference on Information and Knowledge Management, Publishing, 2001, pp. 403–410.

[33] Z. Zhai, B. Liu, H. Xu, P. Jia, Constrained LDA for grouping product features in opinion mining, in: Proceedings of 15th Pacific–Asia Conference, Advances in Knowledge Discovery and Data Mining, 2011, pp 448–459.

[34] T.J. Zhan, C.H. Li, Semantic dependent word pairs generative model for fine-grained product feature mining, Advances in Knowledge Discovery and Data Mining (2011) 460–475.

[35] J. Zhu, H. Wang, M. Zhu, B.K. Tsou, M. Ma, Aspect-based opinion polling from customer reviews, IEEE Transactions on Affective Computing 2 (1) (2011) 37–49.