JIS

Aspect-based sentiment analysis of movie reviews on discussion boards

Tun Thura Thet, Jin-Cheon Na and Christopher S.G. Khoo

Nanyang Technological University, Singapore

Abstract.

In this article, a method for automatic sentiment analysis of movie reviews is proposed, implemented and evaluated. In contrast to most studies that focus on determining only sentiment orientation (positive versus negative), the proposed method performs fine-grained analysis to determine both the sentiment orientation and sentiment strength of the reviewer towards various aspects of a movie. Sentences in review documents contain independent clauses that express different sentiments toward different aspects of a movie. The method adopts a linguistic approach of computing the sentiment of a clause from the prior sentiment scores assigned to individual words, taking into consideration the grammatical dependency structure of the clause. The prior sentiment scores of about 32,000 individual words are derived from SentiWordNet with the help of a subjectivity lexicon. Negation is delicately handled. The output sentiment scores can be used to identify the most positive and negative clauses or sentences with respect to particular movie aspects.

Keywords: discussion board; opinion mining; sentiment analysis

1. Introduction

With the recent proliferation of Web 2.0 sites and applications, and the rapid growth of user-generated content on the internet, many organizations are carrying out sentiment analysis and opinion mining of online review postings. Analysing opinions expressed on various web platforms is increasingly important for effective organizational decision making [1]. Sentiment analysis is a type of text analysis, under the broad area of natural language processing, computational linguistics and text mining [2], that analyses sentiment in a given textual unit with the objective of understanding the polarities of the opinions expressed and the types of emotions toward various aspects of a subject. Sentiments, such as opinions, attitudes, thoughts, judgements and emotions, are private states of individuals which are not open to objective observation or verification. They are expressed in language using subjective expressions [3].

Most of the early studies were focused on binary classification of positive and negative sentiments to predict an overall sentiment of a review document. Recently researchers have been working on

Correspondence to: Jin-Cheon Na, Division of Information Studies, Wee Kim Wee School of Communication and Information, Nanyang Technological University, Singapore 637718. Email: tjcna@ntu.edu.sg

Journal of Information Science, 36 (6) 2010, pp. 823–848 © The Author(s), 2010. Reprints and Permissions: sagepub.co.uk/journalsPermissions.nav, DOI: 10.1177/0165551510388123

aspect-based sentiment analysis which performs more in-depth sentiment analysis of review texts. For instance, when looking at the reviews of a folk song, the opinions in the review text cover not only the overall sentiment but also many specific aspects such as vocals, lyric, recording quality, creativity, and so on. Our study focuses on movie reviews because of the large amount of user-generated reviews available on the internet and the challenges this type of text presents. Sentiment analysis of movie reviews is considered more challenging than sentiment analysis of other types of reviews, such as product reviews [4, 5]. For instance, in movie reviews, an evil character or a tragic storyline does not make a movie bad, but in product reviews, bad product parts usually add up to a bad product [4]. Our study examines not only the overall sentiment of a movie review, but also the sentiment towards various aspects of a movie, such as cast, director, story and music. Separate consideration for the different aspects of a movie review provides better sentiment analysis. For example, the sentence '*I love the story but not the music*' contains two opposing sentiments for two different aspects of the movie – story and music. Our study also investigates the sentiment intensity (i.e. sentiment scores) of clauses or sentences. For example, the sentence '*this is superb*' has a higher sentiment score than the less positive sentence '*this is good*'. Sentiment scores are used to classify the sentiment polarity (i.e. positive, negative or neutral) of clauses or sentences. Additionally, they can be used for comparing one clause or sentence with others in sentiment strength, identifying the most positive and negative clauses or sentences associated with particular aspects, or ranking clauses or sentences in descending sentiment score order.

In order to determine the sentiment intensity and polarity for different aspects of a movie, the proposed approach performs sentiment analysis at the clause level using a linguistic approach. The approach makes use of both a domain-specific lexicon and a generic opinion lexicon, derived from SentiWordNet [6] and the subjectivity lexicon of [7], to assign a prior sentiment score to each word in a sentence. A set of grammatical dependencies between the words in a given sentence, based on the syntactic structure, is generated and a dependency tree is constructed. The dependency tree is divided into dependency sub-trees, each representing a single clause focusing on just one aspect of the movie. Then, the contextual sentiment score of each clause is inferred using the grammatical dependencies and prior sentiment scores of the individual terms. For instance, the sentence '*I love the story but not the music*' is divided into '*I love the story*' and '*I do not love the music*'. After dividing the sentence into separate clauses, a contextual sentiment score toward each movie aspect (e.g. story or music aspect) is calculated. After calculating the contextual sentiment score for each clause, the contextual sentiment score for each review aspect as well as the overall sentiment score for the whole sentence is calculated.

The rest of this paper surveys related studies, describes our approach for clause level sentiment analysis with detailed explanation and reports experimental results with a discussion.

2. Related work

Sentiment analysis is a type of subjectivity analysis which aims to identify opinions, emotions and evaluations expressed in natural language [8]. The main goal is to predict the sentiment orientation (i.e. positive, negative or neutral) by analysing sentiment or opinion words and expressions in sentences and documents. It has been studied by many researchers in recent years.

Sentiment analysis approaches often require resources such as sentiment lexicons to determine which words or phrases are positive or negative in a general context. General Inquirer [9] is a manually compiled resource often used in sentiment analysis. Automatic acquisition of the polarity of sentiment words and phrases is pioneered by the work of Hatzivassiloglou and McKeown [10] who used a machine learning method to construct a lexicon of sentiment terms. Many techniques have been proposed for learning the polarity of sentiment words. Wiebe [11] and Turney [5] studied the extraction of adjectives and adjectival phrases, whereas Riloff, Wiebe and Wilson [12] focused on nouns while Riloff and Wiebe [13] extracted linguistic patterns from subjective expressions. Qiu et al. [14] proposed a propagation approach for extracting a large number of sentiment words and assigning polarity.

Most of the early studies were focused on document level analysis for assigning the sentiment orientation of a document, for example [5, 13, 15–22]. Pang, Lee, and Vaithyanathan [18] used three machine learning methods: naïve Bayes, maximum entropy classification, and support vector machines (SVM), and found that SVM [23] performed relatively better than the other machine learning approaches. Mullen and Collier [16] introduced a hybrid SVM approach by making use of information such as favourability measures of terms and knowledge of the topics. Pang and Lee [17] applied text categorization techniques only to the subjective portions of the document. Their proposed minimum-cut framework improved both the efficiency and accuracy of sentiment classification by utilizing the relation between subjectivity detection and polarity classification. Whitelaw, Garg and Argamon [19] proposed a method for sentiment classification by extracting and analysing appraisal groups, based on the appraisal theory framework [24]. They used semi-automated methods to construct a lexicon of appraising adjectives and their modifiers, and used them as features, together with the standard bag-of-words features, for categorizing movie review documents. However, these document level sentiment analysis approaches are less effective when in-depth sentiment analysis of review texts is required. Different review genres, such as critic reviews, blog posts, message posts on discussion boards and Twitter posts (tweets) have different characteristics and document level sentiment analysis using a bag-of-words approach may be suitable for some genres with long texts, but it is not ideal for other genres having rather short texts [25].

More recently researchers have carried out sentence level sentiment analysis to examine and extract opinions toward various aspects of a reviewed object. In most cases, a sentence level sentiment analysis is more sophisticated than a document level one, and representative works include [26–35]. Hu and Liu [28] mined and summarized customer reviews of electronic products, such as digital cameras, cellular phones, and mp3 players. They extracted the features or aspects (such as picture quality and screen size) of the product on which the customers have expressed their opinions, and predicted whether each opinion sentence is positive or negative. If positive or negative opinion words prevail, the opinion sentence is predicted as positive or negative. Blair-Goldensohn et al. [26] presented a system that summarizes the sentiment of reviews for a local service such as a restaurant, department store or hotel. The set of service reviews associated with a local service returned by Google Maps (maps.google.com) was used as input data. The system extracted relevant aspects of a service, such as service, ambiance, or value, aggregated the sentiment per aspect, and showed aspect-relevant text with sentiment polarity values. They used both lexicon-based and maximum entropy [36] approaches to classify each sentence as positive, negative or neutral.

Yi et al. [33] proposed a method for sentiment analysis of online reviews and news articles to extract positive and negative sentiments for specific subjects from a sentence. They used semantic analysis with a syntactic parser and sentiment lexicons for polarity classification. Miyoshi and Nakagami [31] proposed another linguistic approach for sentiment classification of customer reviews on electronic products. Adjective–noun pairs were analysed, taking into consideration the contextual valence shifters that change the semantic orientation. Shaikh, Prendinger and Ishizuka [32] proposed a sentence level sentiment analysis approach where a linguistic tool called SenseNet was developed for domain independent sentiment analysis and visualization of the results. The approach analysed the semantic verb frames of a sentence to calculate the contextual valence (i.e. score) of a whole sentence, but did not compute separate contextual scores for multiple aspects discussed in a sentence. Zhang et al. [34] performed sentiment analysis of Chinese content rather than English. They used a rule-based approach to determine a sentence's sentiment based on word dependency, and then to predict a document sentiment by aggregating the analysis results of individual sentences. They assigned sentences different weights to adjust their contribution to the overall sentiment polarity of a document.

Ding, Liu and Yu [27] proposed a sentence level lexicon-based approach which is closely related to our method (although the study focused on sentence level analysis, it treated a short passage containing a sequence of several sentences as a sentence). They studied the problem of determining binary-valued sentiment orientation of opinions toward product features (or aspects), but did not attempt to assign sentiment scores. The sentiment orientation of a feature in a given sentence was

calculated based on the occurrence of sentiment words which are assigned +1 for positive and –1 for negative. The approach also considered distance between a feature word and a sentiment word, and used a linguistic parser to determine the part-of-speech (POS) tags of the words. However, the grammatical dependencies of words in the sentences were not considered. It also handled negation of words by recognizing known patterns. Since it focused only on sentiment orientation, it did not handle negation delicately. For instance, '*not great*' is less negative than '*not good*' whereas '*not true*' is more negative than '*hardly true*'. The approach performed sentiment summarization based on the number of positive opinions versus the number of negative opinions, but it did not consider how positive or negative the opinions were.

In contrast to most studies which focused on document level or sentence level sentiment analysis, our work studies clause level sentiment analysis so that different opinions on multiple aspects expressed in a sentence can be processed separately in each clause. Some researchers have studied phrase level contextual sentiment analysis, but phrases are often not long enough to contain both sentiment and feature terms together for detailed analysis [7, 37]. The study that is closest to ours is that by Wilson, Wiebe and Hwa [38] who identified opinions in clauses and classified their intensities. They used three machine learning approaches, BoosTexter [39], Ripper [40] and SVM [23], and investigated a wide range of clause features, including syntactic and lexical clues generated from dependency parse trees. Each sentence is divided into nested clauses, with the top level clause representing the entire sentence. Each clause is classified to one of the intensity values (neutral, low, medium or high). Their study is similar to ours in that the opinion intensity of clauses (and sentences) is analysed using grammatical relationships of words. However, in their study, the grammatical relationships are used as features in the feature vectors that represent the clauses and sentences. The machine learning methods are applied to the feature vectors for classifying opinion intensity values. In contrast, we use a linguistic approach in which grammatical relationships are used with refined rules to calculate contextual sentiment scores of clauses. In addition, Wiebe et al. did not look into the sentiment polarity and aspect of each clause, and the intensity scale used is ordinal-valued (i.e. neutral, low, medium or high), whereas our approach classifies the sentiment polarity of each clause and tags its aspect, and the intensity scale is real valued (between –1 and +1). Our linguistic approach uses a more refined sentiment score calculation method to determine the sentiment intensity of a clause or sentence, and to compare it with others in terms of sentiment strength. For example, the proposed approach can determine that '*this movie is very good*' is more positive than the clause '*this movie is fairly good*' because the adverb '*very*' having a higher prior sentiment score than '*fairly*' intensifies the positive adjective '*good*' using the defined rule. In addition, the two clauses are determined as referring to the overall aspect because of the term '*movie*'. It also handles negation delicately. For instance, '*not outstanding*' is less negative than '*not good*', and *'cease boring*' becomes negative because of the negating term '*cease*'. The sentence level sentiment score is also calculated by averaging the sentiment scores for all aspects, whereas Wilson et al. used a maximum intensity value in a sentence without considering aspects. More details of our approach are discussed in the following sections.

3. Sentiment analysis

3.1. Overview

Generally, there are three types of sentences in the English language: simple sentence, compound sentence and complex sentence. Simple sentences contain only one independent clause whereas compound sentences contain two or more independent clauses, and complex sentences contain at least one independent clause and one dependent clause. In structure, a clause comprises a subject and a predicate, where the predicate is a combination of verb, object, complement and adverbial [3]. The subject is usually a noun phrase that names a person, place or thing. The verb identifies an action or a state of being. An object receives the action and usually follows the verb. There are basically seven types of clauses as described in Table 1.

Table 1
Clause structures

			Predicate			
		Subject	Verb phrase	Object	Complement	Adverbial
1	SV	*He*	*smiled*			
2	SVO	*He*	*likes*	*music*		
3	SVC	*He*	*became*		free	
4	SVA	*He*	*is*			*in the room*
5	SVOO	*He*	*gave*	*her a car*		
6	SVOC	*He*	*consider*	*this*	*rather expensive*	
7	SVOA	*He*	*parked*	*the car*		inside

```
FOR each sentence
    Perform semantic annotation
    Assign prior sentiment scores to words
    Generate grammatical dependencies
    Break a sentence into clauses
    FOR each clause
        Calculate clause level sentiment score (Single Aspect)
        Subject or Object (Adjective + Noun)
        Verb phrase (Adverb + Verb)
        Predicate (Verb phrase + Object/Complement)
        Clause (Subject and Predicate)
        Determine Aspect
    END FOR
    Calculate sentence level sentiment score (Multiple Aspect)
END FOR
```

Fig. 1. Algorithm for clause level sentiment analysis.

Figure 1 gives an overview of the algorithm used for clause level sentiment analysis of movie review texts. Firstly, for each sentence of the review text, semantic annotation is performed, and prior sentiment scores are assigned to each word. Then, the grammatical dependencies are determined, and the sentence is broken into independent clauses. For each clause, the contextual sentiment score is calculated by traversing the dependency tree based on its clause structure. The review aspect of the clause is determined by locating an occurrence of a movie aspect feature word. After processing all the clauses of the sentence, the sentiment score for each review aspect is calculated by taking the average of the clauses addressing the same aspect. The sentiment score for the whole sentence is determined from the calculated scores of multiple aspects.

3.2. Semantic annotation

Movie names (e.g. '*beautiful mind*' and '*fantastic four*') sometimes contain sentiment words. These words (i.e. *'beautiful*' and *'fantastic*') should not be processed for sentiment analysis. The objective of semantic annotation is to tag the text with semantic categories, such as movie, director, cast and character names, so that the tagged terms can be ignored during sentiment processing. These tags will also be useful when determining review aspects. Data about each movie (i.e. movie, director, cast and character names) are collected from The Internet Movie Database (www.imdb.com) and stored in a movie specific feature list.

In addition, the words indicating movie aspects such as '*direct*', '*animation*', '*scene*', '*music*' and '*sound effects*' are also prepared and stored in the movie aspect feature list. The aspect terms are collected from the dataset of 520 movie reviews used in our previous study [25], which are from various online genres (critic reviews, user reviews, discussion boards and blogs) of 38 movies (this dataset is different from the dataset used for evaluation in this study). From this dataset, sentences

Table 2
Movie aspect and movie specific feature terms

Aspect	Movie aspect Feature terms	Movie specific Feature terms (metadata)
Overall	*movie, film*	<Movie Names>
Cast	*act, acting, actress, actor, role, portray, character, villain, performance, performed, played, casting, cast*	<Actor Names> <Character Names>
Director	*direct, direction, directing, director, filmed, filming, filmmaking, filmmaker, cinematic, edition, cinematography*	<Director Names>
Story	*storyline, story, tale, romance, dialog, script, storyteller, ending, storytelling, revenge, betrayal, plot, writing, twist, drama*	<Script Writers>
Scene	*scene, scenery, animation, violence, screenplay, action, special effect, stunt, shot, visual, props, camera, graphic*	<Animators> <Cameramen>
Music	*lyric, sound, music, audio, musical, title track, sound effect, sound track*	<Sound Tracks>

or paragraphs covering each review aspect are read by one of the authors and important feature terms specific to each aspect are manually extracted. Annotation is performed by tagging the terms that matched the ones in the movie aspect and movie specific feature lists. For the matching, firstly, terms in sentences are stemmed and tagged with part-of-speech (POS), and they are compared with the entries with the same POS tags (each entry in the movie aspect feature list contains an associated POS). For instance, the verb entry '*act*' in the list matches any form of the verb '*act*' (e.g. '*acts*' or '*acted*'), but does not match the noun term '*act*'. Similarly, the noun entry '*sound*' matches either singular or plural noun (e.g. '*sounds*'), but not the verb term '*sound*'. The longest matching method is applied, and each entry in the feature lists is given a unique annotation code. For POS tagging, the Stanford Log-linear Part-Of-Speech Tagger is used [41].

Table 2 gives sample entries of the movie aspect and movie specific feature lists which are used for annotation. The phrase '*fantastic four*' in the sentence '*fantastic four is a boring movie.*' is annotated with a movie name tag to avoid processing the positive word '*fantastic*' during sentiment analysis. The following is an example of an annotated sentence:

<MovieName:F0001> fantastic four </MovieName:F0001> is a boring movie.

3.3. Prior sentiment scores

The prior sentiment scores of the individual words in an input clause are used to calculate the contextual sentiment score of the clause. The prior sentiment scores of the words range between –1 and +1, with 0 being neutral. In the study, two lexicons, a domain specific lexicon and a generic opinion lexicon, are created to store the prior sentiment scores of opinion words. The domain specific lexicon contains movie domain specific opinion words, and the generic opinion lexicon holds general opinion words derived from SentiWordNet and the subjectivity lexicon.

3.3.1. Domain specific lexicon

We have prepared a set of domain specific opinion words for the movie review domain by semi-automatically analysing the movie reviews in the separate training dataset (i.e. the dataset of 520 movie reviews). Firstly, words and their information gain measure are extracted and calculated from the training dataset. Information gain is frequently used as a term goodness measure in the areas of machine learning and information retrieval [42, 43]. Yang and Pedersen [44] performed a comparative study of feature (i.e. important word) selection methods in statistical learning of text categorization. They found information gain and chi-square statistic most effective in their experiments. Therefore, information gain is used in this study to extract important opinion words strongly associated with positive or negative movie reviews. We also experimented with the chi-square statistic and obtained similar results.

Table 3
Domain specific word list

Terms	Domain specific	Subjectivity/SentiWordNet
rock	+0.2357	0
suck	–0.23375	0
forgettable	–0.23375	+0.375
magical	+0.2357	–0.25
act	0	+0.0375
scene	0	–0.0625

The following formula is used for calculating the information gain of a word, which measures the expected reduction in entropy caused by partitioning the training documents according to the word:

$$IG(w) = \Pr(w) \bullet \sum_{i=1}^{K} \Pr(c_i \mid w) \log \frac{\Pr(c_i \mid w)}{\Pr(c_i)} + \Pr(\bar{w}) \bullet \sum_{i=1}^{K} \Pr(c_i \mid \bar{w}) \log \frac{\Pr(c_i \mid \bar{w})}{\Pr(c_i)} \quad (1)$$

where w is a word, K the total number of classes (i.e. two in our dataset), c_1 and c_2 the positive and negative categories, $\Pr(c_i)$ the percentage of training documents in category c_i, $\Pr(w)$ the percentage of training documents in which word w is present, $\Pr(\bar{w})$ the percentage of documents in which word w is absent, $\Pr(c_i \mid w)$ is the conditional probability of category c_i given word w, and $\Pr(c_i \mid \bar{w})$ is the conditional probability of category c_i given that word w is absent.

Then opinion words with high information gain are manually examined and added to the domain specific lexicon if they are found to be domain specific. We have added about 100 words to the domain specific lexicon. For instance, although the word '*unpredictable*' is considered negative in a general context (i.e. in the subjectivity and SentiWordNet lexicons), it often reflects positive sentiment for a movie storyline. Some informal words such as '*sucks*' and '*rocks*' are frequently used in movie reviews with specific sentiment polarities and they are added to the domain specific lexicon. Some words have positive or negative sentiment scores in the general context, but they are used without sentiment meaning in the movie review domain. For example, the words '*act*' and '*scene*' are added as neutral words in the domain specific list. Table 3 shows sample entries of domain specific opinion terms. The positive and negative scores for the domain specific terms are set to the average sentiment values (i.e. the K value +0.2357 for positive terms and the M value –0.23375 for negative terms in Table 4). This will be discussed in more detail in the generic opinion lexicon section.

3.3.2. Generic opinion lexicon

SentiWordNet (http://sentiwordnet.isti.cnr.it/) is mainly used to collect general opinion words and derive their prior sentiment scores. Each word in the generic opinion lexicon has a prior sentiment score which ranges between –1 and +1. SentiWordNet is a lexical resource for sentiment analysis, and its sentiment scores were automatically calculated using a semi-supervised method described in [6]. Each synset of WordNet is assigned with positive, negative and objective scores. In SentiWordNet, there are over 200,000 entries describing sentiment scores for multiple senses of words and phrases. For example, as shown in Figure 2, one of the senses of the word '*great*' has the sentiment scores of positive 0.375, negative 0.125 and objective 0.5.

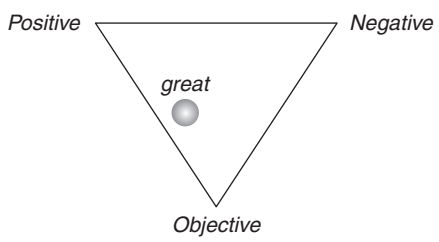

Fig. 2. SentiWordNet.

Since our approach ignores word senses in sentences, the positive and negative values of the multiple senses of a word in SentiWordNet are converted to one representative positive or negative value. In addition, our approach uses only the unigram word entries in SentiWordNet since the sentiment scores of phrases in our study are calculated based on the grammatical relationships between the constituent words in the phrases.

To construct the generic opinion lexicon, each word in SentiWordNet is added to the lexicon. Then the subjectivity lexicon of [7] is used to determine the sentiment orientation of the terms in the generic opinion lexicon (as it was found to be more accurate than SentiWordNet). The subjectivity lexicon contains subjective words which can be used to express private states or sentiments. The words which are subjective to most contexts are classified as either *strongly positive* or *strongly negative*, and the words which may only have certain subjective usage are classified as *weakly positive* or *weakly negative*. The words were extracted and expanded from General Inquirer and the work of Hatzivassiloglou and McKeown [10] and Riloff et al. [12]. The size of the subjectivity lexicon is about 8000 words where 33% are positive, 60% negative and 7% either neutral or both.

We use the positive and negative entries in the subjectivity lexicon to determine the overall orientation of the words in the generic opinion lexicon. The prior scores are calculated based on the scores of their senses in SentiWordNet. Initially, each term in the generic opinion lexicon is classified as positive or negative by taking the average score of all the senses of the term in SentiWordNet. If the average score is larger than 0, it becomes positive, otherwise negative (note that all the terms in the generic opinion lexicon are grouped by polarity in Table 4). Table 4 summarizes how the prior sentiment scores of the words in the generic opinion lexicon are derived from the subjectivity lexicon and SentiWordNet. The rationale for the method is described with examples in the following paragraph. The computed prior sentiment scores are used to calculate sentiment scores of clauses and sentences, and their effectiveness is evaluated and reported in Section 4.

When a term in the generic opinion lexicon is strongly positive (A and F in Table 4) in the subjectivity lexicon, the maximum value of the positive scores of its multiple senses in SentiWordNet is used as a prior sentiment score of the term. For instance, the term '*great*' has multiple senses with positive scores (+0.25, +0.357, +0.5 and +0.625) in SentiWordNet and it is strongly positive in the subjectivity lexicon. Thus, its prior score is set to the maximum value (+0.625) since this term is considered to be strongly positive in most contexts. Similarly, when a term is strongly negative (C and H in Table 4) in the subjectivity lexicon, the minimum value of its negative scores is used as a prior sentiment score of the term. When a term is weakly positive (B and G in Table 4) in the subjectivity lexicon, the average value of its positive scores is used as a prior sentiment score of the term since the term is subjective only in certain usage, and the

Table 4
Derivation of sentiment scores using the subjectivity lexicon and SentiWordNet

		Subjectivity				
		Strongly positive	Weakly positive	Strongly negative	Weakly negative	NA
Generic opinion lexicon	**Positive**	A	B	C	D	E
	Negative	F	G	H	I	J
	NA	K	L	M	N	
Formula						
A, F	= *Max* (positive scores of multiple senses)					
B, G, E	= *Avg* (positive scores of multiple senses)					
C, H	= *Min* (negative scores of multiple senses)					
D, I, J	= *Avg* (negative scores of multiple senses)					
K	= (*Avg* (scores of all instances of A) + *Avg* (scores of all instances of F))/2					
L	= (*Avg* (scores of all instances of B) + *Avg* (scores of all instances of G))/2					
M	= (*Avg* (scores of all instances of C) + *Avg* (scores of all instances of H))/2					
N	= (*Avg* (scores of all instances of D) + *Avg* (scores of all instances of I))/2					

```
IF ( Term in [Domain Specific Lexicon]) THEN
    Prior Score = The Domain Specific Score of the Term;
ELSE IF ( Term in [Generic Opinion Lexicon] ) THEN
    Prior Score = The Generic Opinion Score of the Term;
ELSE
    Prior Score = Neutral;
END IF
```

Fig. 3. Procedure for assigning prior sentiment scores.

maximum or minimum value of the positive scores of its various senses may reflect only special usages of the term. For instance, the term '*satisfying*' has multiple senses with positive scores (+0.38, +0.38 and +0.63) and it is weakly positive in the subjectivity lexicon. Thus, its prior score is set to the average value (+0.46). Similarly, when a term is weakly negative (D and I in Table 4) in the subjectivity lexicon, the average value of all the negative scores is used as a prior sentiment score of the term. When a term is not found in the subjectivity lexicon but in the generic opinion lexicon (E and J in Table 4), the overall sentiment orientation cannot be determined by the subjectivity lexicon. Then the average value of either positive or negative scores of multiple senses of the term becomes its prior score, depending on its overall orientation determined by taking the average score of all the senses of the term in SentiWordNet. When a term is not found in the generic opinion lexicon but in the subjectivity lexicon (K, L, M and N in Table 4), then a prior score is assigned using the associated formula K, L, M or N for strongly positive, weakly positive, strongly negative or weakly negative cases, respectively. This is the case where additional words are added to the generic opinion lexicon from the subjectivity lexicon. Generally the scores of the terms in K, L, M and N are inferred from the scores of the other terms occurring in both the generic opinion lexicon and subjectivity lexicon. For instance, when a term (K in Table 4) is not found in the generic opinion lexicon but strongly positive in the subjectivity lexicon, its prior score is the midpoint between the average prior score of all the terms in A and the average prior score of all the terms in F. Consequently, by considering all the strongly positive terms occurring in the generic opinion lexicon, the terms in K become more positive than the terms in F, but less positive than the terms in A. In our final generic opinion lexicon, there are around 32,000 word entries assigned with prior sentiment scores.

The procedure in Figure 3 shows the overall steps for assigning a prior sentiment score to each word in a clause or sentence before traversing the dependency tree to calculate the contextual sentiment score of the clause or sentence. As shown in the algorithm, the domain specific lexicon has higher priority than the generic opinion lexicon for assigning a prior sentiment score.

3.4. Dependency tree

We have used the Stanford NLP library [45] to process the grammatical relationships of words in a sentence. There are 55 Stanford type dependencies [46]. Table 5 shows some of the commonly used Stanford type dependencies for sentiment analysis.

The Stanford type dependencies are binary grammatical relationships between two words: a governor and a dependent. For example, the sentence '*I love the story but not the music*' has the following dependencies among the words.

- nsubj[love-2, I-1];
- dobj[love-2, story-4];
- det[story-4, the-3];
- conj_negcc[story-4, music-8];
- det[music-8, the-7].

Table 5
Sample grammatical relationship types

Dependency code	Description
nsub	Nominal subject
nsubpass	Passive nominal subject
csub	Clausal subject
csubpass	Passive clausal subject
dobj	Direct object
amod	Adjective modifier
acomp	Adjective complement
advmod	Adverbial modifier
advcl	Adverbial clause modifier
conj	Conjunction
xcomp	Open clause complement
prt	Phrasal verb particle
conj_negcc	Conjunction/negation
det	Determiner

In the word dependencies, '*nsubj[love-2, I-1]*' indicates that the first word '*I*' is a nominal subject of the second word 'love'. In the dependency relationship, 'I' is the dependent and '*love*' is the governor. '*dobj[love-2, story-4]*' indicates that the fourth word '*story*' is the direct object of the governor '*love*'. '*det[story-4, the-3]*' and '*det[music-8, the-7]*' indicate that '*the*' in the third and seventh position is a determiner of '*story*' and '*music*' respectively. '*conj_negcc[story-4, music-8]*' indicates that '*story*' and '*music*' are connected by a coordinating conjunction with a negation word.

Based on the output set of grammatical dependencies, a dependency tree is constructed as shown in Figure 4. For better performance, the dependency tree is trimmed by removing less important grammatical dependencies for sentiment analysis. Dependencies such as *determiner* are removed from the tree since they are not useful for sentiment analysis.

The whole sentence is represented by a dependency tree which can be divided into sub-trees, where each sub-tree represents a clause focusing on a particular aspect. Division of a dependency tree (the whole sentence) into sub-trees (clauses) is performed by one of two methods: splitting or splitting with replication.

A compound sentence with two or more independent clauses with complete clause structure can be divided by splitting them directly. For example, the sentence '*I love Tom but I hate the movie*' can be divided simply into '*I love Tom*' (cast aspect) and '*I hate the movie*' (overall aspect) as shown in Figure 5, so that its sentiment can be analysed more accurately for each aspect.

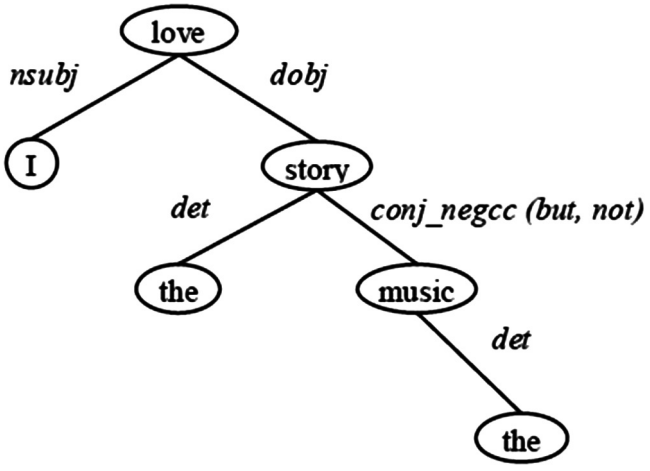

Fig. 4. Graphical representation of the dependency tree for the sentence '*I love the story but not the music*'.

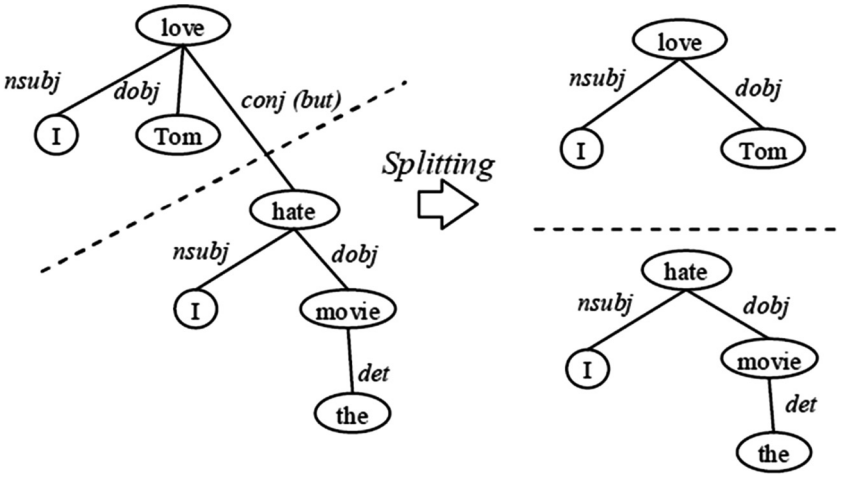

Fig. 5. Dividing the dependency tree for the sentence '*I love Tom but I hate the movie*'.

For the following sentence structures with compound predicates or compound objects, some portion of the tree needs to be replicated since they cannot be split directly:

- Subject1 Verb1 and Verb2 Object1;
- Subject1 Verb1 Object1 and Object2;
- Subject1 Verb1 Object1 but not Object2.

For example, the sentence '*I love the music but not the story*' needs to be split into '*I love the music*' and '*negation(I love the story)*' as shown in Figure 6, in order that their sentiments can be analysed independently. As discussed before, in the actual processing, dependencies such as *determiner* (i.e. the) are trimmed to reduce noise and improve efficiency.

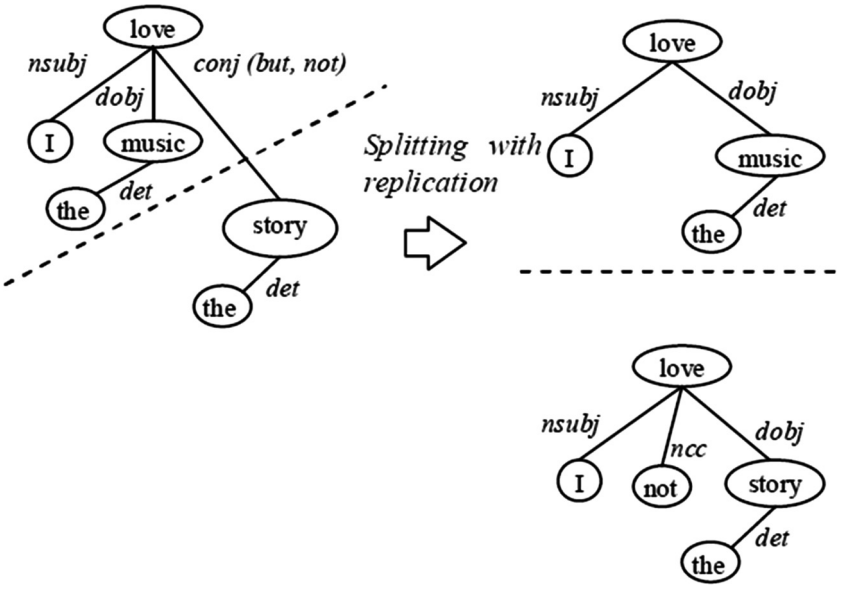

Fig. 6. Dividing the dependency tree for the sentence '*I love the music but not the story*'.

3.5. Contextual sentiment score

A previous study by Shaikh et al. [32] proposed a similar approach of semantic dependency analysis for sentence-level sentiment. In our proposed approach, the analysis is performed at the clause level for aspect-level sentiment, and more refined calculation rules are used to provide more accurate sentiment scores by using the grammatical dependences, sentiment polarity, part-of-speech and prior sentiment scores of terms in the clause. The calculation rules and formulas for processing contextual sentiment scores will be discussed in the following subsections.

3.5.1. Subject or object

The subject or object of a clause can be a noun phrase consisting of a noun and an adjective. When inputs (i.e. adjective and noun) are of the same sentiment orientation (F1.1 and F1.4 in Table 6), they tend to intensify each other. The absolute value of the output should be larger than the absolute values of the inputs but less than 1. Therefore, the formula (|N| + (1–|N|) * |A|) is applied. For example, the adjective '*great*' in the phrase '*great art*' intensifies the positive noun '*art*'. Thus, the phrase '*great art*' becomes more positive than the noun '*art*' itself. The prior sentiment scores of the noun '*art*' and the adjective '*great*' are +0.125 and +0.625 respectively. The output sentiment score of the phrase '*great art*' is calculated as (+ (0.125 + (1 – 0.125) * 0.625) = +0.671) which is more positive than the original scores +0.125 and +0.625. Similarly, the adjective '*worst*' in the phrase '*worst disaster*' intensifies the negative noun '*disaster*'. Thus, the phrase '*worst disaster*' becomes more negative than the noun '*disaster*' itself. The prior sentiment score of the noun '*disaster*' and the adjective '*worst*' are –0.5 and –0.75 respectively. The output sentiment score of the phrase '*worst disaster*' is calculated as (–(0.5 + (1–0.5) * 0.75) = –0.875) which is more negative than the original scores –0.5 and –0.75.

When the adjective is positive and the noun is negative (i.e. F1.2 in Table 6), the adjective also intensifies the noun. For example, the adjective '*big*' in the phrase '*big failure*' intensifies the noun '*failure*'. Thus, the phrase '*big failure*' becomes more negative than the noun '*failure*' itself. The prior sentiment scores of the noun '*failure*' and the adjective '*big*' are –0.214 and +0.225 respectively. The output sentiment score of the phrase '*big failure*' is calculated as (–(0.214 + (1–0.214) * 0.225) = –0.39) which is more negative than the original score of 'failure' –0.214. When the adjective is negative and noun is positive (i.e. F1.3 in Table 6), the output is the value of the negative adjective. For example, the adjective '*lousy*' in the phrase '*lousy star*' determines the output score. The prior sentiment scores of the adjective '*lousy*' and the noun '*star*' are –0.75 and +0.125 respectively. Thus, output sentiment score of the phrase '*lousy star*' becomes –0.75.

Table 6
Contextual sentiment for subject or object

ID	(A)djective	(N)oun	Output	Examples
F1.1	+ / 0*	+ / 0	+	great art
F1.2	+ / 0	–	–	big failure
F1.3	–	+ / 0	–	lousy star
F1.4	–	–	–	worst disaster
Formula				
F1.1	Positive A and Positive N (positive includes neutral) ⇒ + (|N| + (1–|N|) * |A|) E.g. great art: +0.625 and +0.125 ⇒ + (0.125 + (1 – 0.125) * 0.625) = +0.671			
F1.2	Positive A and Negative N ⇒ – (|N| + (1–|N|) * |A|) E.g. big failure: +0.225 and –0.214 ⇒ – (0.214 + (1 – 0.214) * 0.225) = –0.39			
F1.3	Negative A and Positive N ⇒ A E.g. lousy star: –0.75 and +0.125 ⇒ –0.75			
F1.4	Negative A and Negative N ⇒ – (|N| + (1–|N|) * |A|) E.g. worst disaster: –0.75 and –0.5 ⇒ – (0.5 + (1 – 0.5) * 0.75) = –0.875			

* 0 indicates neutral.

3.5.2. Verb phrase

A verb phrase can contain a verb and an adverb. Similar formulas are used (F2.1, F2.2, F2.3 and F2.4 in Table 7). For the negating adverb such as '*hardly*' and '*rarely*', the negation method (F2.5 in Table 7) is used. The negating adverb can shift the original sentiment orientation of a verb or an adjective. When a verb is positive, the result becomes negative, and vice versa. For example, in the verb phrase '*rarely pass*', negating the positive verb '*pass*' with '*rarely*' makes the phrase negative. However, the phrase '*rarely pass*' is not as negative as the phrase '*not pass*' and thus the formula +/– (|A| * (1 – |V|)) is used.

3.5.3. Predicate

A predicate can contain a verb phrase and an object or complement. Similar formulas are used (F3.1, F3.2, F3.3 and F3.4 in Table 8). For the negating verbs, the negation method (F3.5 in Table 8) is used.

Table 7
Contextual sentiment for verb phrase

ID	(A)dverb	(V)erb	Output	Examples
F2.1	+ / 0	+ / 0	+	cheer happily
F2.2	+ / 0	–	–	gossip proudly
F2.3	–	+ / 0	–	liberate wrongly
F2.4	–	–	–	fail badly
F2.5	–	–	+	hardly fail (*negation exception)
	–	+	–	rarely pass
Formula				
F2.1	Positive A and Positive V ⇒ + (|V| + (1–|V|) * |A|) E.g. cheer happily: +0.5(A) and +0.625(V) ⇒ + (0.625 + (1 – 0.625) * 0.5) = +0.812			
F2.2	Positive A and Negative V ⇒ – (|V| + (1–|V|) * |A|) E.g. gossip proudly: +0.375(A) and –0.625(V) ⇒ – (0.625 + (1 – 0.625) *0.375) = –0.765			
F2.3	Negative A and Positive V ⇒ A E.g. liberate wrongly: –0.75(A) and +0.375(V) ⇒ –0.75			
F2.4	Negative A and Negative V ⇒ – (|V| + (1 – |V|) * |A|) E.g. fail badly: –0.437(A) and –0.068(V) ⇒ – (0.068 + (1 – 0.068) * 0.437) = –0.475			
F2.5	Negative A and Negative V ⇒ + (|A| * (1 – |V|) E.g. hardly fail: –0.6(A) and –0.5(V) ⇒ + (0.6 * (1 – 0.5)) = +0.3 Negative A and Positive V ⇒ – (|A| * (1 – |V|) E.g. rarely pass: –0.6(A) and 0.25(V) ⇒ – (0.6 * (1 – 0.25)) = –0.45			

Table 8
Contextual sentiment for predicate

ID	(V)erb Phrase	(O)bject/Complement	Output	Examples
F3.1	+ / 0	+ / 0	+	provide goodness
F3.2	+ / 0	–	–	provide problems
F3.3	–	+ / 0	–	lost award
F3.4	–	–	–	suffers pain
F3.5	–	–	+	ceased boring (*negation exception)
	–	+	–	stopped winning
Formula				
F3.1	Positive V and Positive O ⇒ + (|O| + (1 – |O|) * |V|) E.g. provide goodness: +0.035 and +0.875 ⇒ + (0.875 + (1–0.875) * 0.035) = +0.879			
F3.2	Positive V and Negative O ⇒ – (|O| + (1 – |O|) * |V|) E.g. provide problems: +0.035 and –0.184 ⇒ – (0.184 + (1–0.184) * 0.035) = –0.212			
F3.3	Negative V and Positive O ⇒ V E.g. lost award: –0.537 and +0.083 ⇒ –0.537			
F3.4	Negative V and Negative O ⇒ – (|O| + (1 – |O|) * |V|) E.g. suffers pain: –0.815 and –0.75 ⇒ – (0.75 + (1 – 0.75) * 0.815) = –0.954			
F3.5	Negative V and Negative O ⇒ + (|V| * (1 – |O|)) E.g. ceased boring: –0.6 and –0.5 ⇒ + (0.6 * (1 – 0.5)) = +0.3 Negative V and Positive O ⇒ – (|V| * (1 – |O|)) E.g. stopped winning: –0.6 and +0.5 ⇒ – (0.6 * (1 – 0.5)) = –0.3			

3.5.4. Clause

A clause contains a subject and a predicate. As before, when inputs are of the same sentiment orientation, they intensify each other (F4.1 and F4.4 in Table 9). When the subject is positive and the predicate is negative (F4.2 in Table 9), the output is the value of the negative predicate. However, when the subject is negative and predicate is positive (F4.3 in Table 9), the output can be either positive or negative. Therefore, the values of the subject and the predicate are compared, and if the absolute value of the subject is larger than that of the predicate, the output becomes the sentiment score of the subject, and vice versa. For example, in the clause '*Disaster started on time*', the absolute value of the subject '*Disaster*' is greater than that of the predicate '*started on time*'. Thus, the output is the sentiment score of '*Disaster*' which is negative. However, in the clause '*This short film was incredible*', the absolute value of the predicate '*was incredible*' is greater than the value of the subject '*the short film*', and thus the output is the sentiment score of '*was incredible*' which is positive.

3.5.5. Complex clause

For complex sentences with '*to*' dependency, the sentiment score of the second clause is intensified when the first clause is positive (F5.1 and F5.2 in Table 10), and the sentiment score of the second clause is negated when the first clause is negative (F5.3 and F5.4 in Table 10).

Table 9
Contextual sentiment for clause

ID	(S)ubject	(P)redicate	Output	Examples
F4.1	+ / 0	+ / 0	+	The superstar did great
F4.2	+ / 0	–	–	The superstar performed poorly
F4.3	–	+ / 0	+ / –	This short film was incredible Disaster started on time
F4.4	–	–	–	Bad casting spoiled everything
Formula				
F4.1	Positive S and Positive P ⇒ + (|P| + (1 – |P|) * |S|) E.g. +0.125 and +0.625 ⇒ + (0.625 + (1 – 0.625) * 0.125) = +0.671			
F4.2	Positive S and Negative P ⇒ P E.g. +0.125 and –0.75 ⇒ –0.75			
F4.3	Negative S and Positive P ⇒ If |S| ⇒ |P| Then S Else P E.g. –0.358 and 0.625 ⇒ +0.625 –0.5 and +0.1 ⇒ –0.5			
F4.4	Negative S and Negative P ⇒ + (|P| + (1 – |P|) * |S|) E.g. +0.597 and +0.583 ⇒ – (0.583 + (1 – 0.583) * 0.597) = –0.831			

Table 10
Contextual sentiment for complex-to

ID	Clause 1	Clause 2	Output	Examples
F5.1	+ / 0	+ / 0	+	I love to watch this great movie again
F5.2	+ / 0	–	–	I will advise to throw away the ticket
F5.3	–	+ / 0	–	It is hard to like this movie
F5.4	–	–	+	It is hard to find problem
Formula				
F5.1	Positive C1 and Positive C2 ⇒ + (|C2| + (1 – |C2|) * |C1|) E.g. +0.375 and +0.658 ⇒ + (0.658 + (1 – 0.658) * 0.375) = +0.786			
F5.2	Positive C1 and Negative C2 ⇒ – (|C2| + (1 – |C2|) * |C1|) E.g. +0.6 and –0.176 ⇒ – (0.176 + (1 – 0.176) * 0.6) = –0.8			
F5.3	Negative C1 and Positive C2 ⇒ – (|C1| * (1 – |C2|)) E.g. –0.25 and +0.5 ⇒ – (0.25 * (1 – 0.5)) = –0.125			
F5.4	Negative C1 and Negative C2 ⇒ + (|C1| * (1 – |C2|)) E.g. –0.25 and –0.184 ⇒ + (0.25 * (1 – 0.184)) = +0.388			

3.5.6. Default method

Since the previously defined rules cannot comprehensively cover all the grammatical dependencies of words in clauses, the default calculation method defined in Table 11 is applied to unmatched phrases. The formulas are generalized since the output sentiment orientation can vary in such situations. When both terms are either positive or negative, they intensify each other and the output maintains their original sentiment orientation (F6.1 and F6.4 in Table 11). However, when their sentiment orientations are different, the term with a greater sentiment score is used as the output (F6.2 and F6.3 in Table 11). For instance, in the case of '*the film's flaws*', '*film*' (noun) and '*flaws*' (noun) are associated with the possession modifier relation. This case is processed using the formula F6.3 in Table 11.

3.5.7. Negation of term

Handling of negation is one of the key processes in sentiment analysis. For example, in the clause '*this is not good*', the negativity (i.e. sentiment score) of the word '*not*' is –1 and the sentiment score of the term '*good*' is +0.583. The output becomes –0.417 using the formula F7.1 in Table 12. As another example, when the score of '*superb*' is +0.875, the sentiment score for '*this is not superb*' becomes –0.125. Since '*superb*' is more positive than '*good*', '*not superb*' becomes less negative than '*not good*'. Also the negativity of '*hardly*' is –0.75 which is less than '*not*'. So the sentiment score of '*hardly good*' becomes –0.312, which is less negative than the sentiment score of '*not good*'.

Figure 7 illustrates how the algorithm performs sentiment analysis of a sentence using most of the rules discussed previously.

Table 11
Default method for contextual sentiment

ID	Term A	Term B	Output	Examples
F6.1	+ / 0	+ / 0	+	make me feel lucky [clausal complement(make/verb, feel lucky/verb + adjective)]
F6.2	+ / 0	–	+ / –	in terms of tension [preposition *of* (terms/noun, tension/noun)]
F6.3	–	+ / 0	+ / –	the film's flaws [possession(flaws/noun, film/noun)]
F6.4	–	–	–	seemed distracting and inappropriate [conjunction *and* (distracting/adjective, inappropriate/adjective)]
Formula				
F6.1	Positive A and Positive B ⇒ + (|A| + (1 – |A|) * |B|) E.g. +0.0 and +0.882 ⇒ + (0.0 + (1 – 0.0) * 0.882) = +0.882			
F6.2	Positive A and Negative B ⇒ If |A| ⇒ |B| Then A Else B E.g. +0.0 and –0.125 ⇒ –0.125			
F6.3	Negative A and Positive B ⇒ If |A| ⇒ |B| Then A Else B E.g. –0.184 and 0.0 ⇒ –0.184			
F6.4	Negative A and Negative B ⇒ – (|A| + (1 – |A|) * |B|) E.g. –0.233 and –1.0 ⇒ – (0.233 + (1 – 0.233) * 1.0) = –1			

Table 12
Contextual sentiment for negation of term

ID	(N)egation	(T)erm	Output	Examples
F7.1	–	+ / 0	–	Not good
F7.2	–	–	+	Not lousy
Formula				
F7.1	Negation N and Positive Term T ⇒ – (|N| * (1 – |T|)) E.g. Not good: –1 and +0.538 ⇒ – (1 * (1 – 0.538)) = –0.417			
F7.2	Negation N and Negative Term T ⇒ + (|N| * (1 –|T|)) E.g. Not lousy: –1 and –0.75 ⇒ + (1 * (1 – 0.75)) = +0.25			

"I can happily watch this great movie to enjoy everything."
movie (0.0) + *great* (0.625)
F1.1-Subject/Object: 0.625
watch (0.089) + *happily* (0.5)
F2.1-Verb Phrase: 0.544
Verb Phrase (0.544) + Subject/Object (0.625)
F3.1-Predicate: 0.829
I (0.0) + Predicate (0.829)
F4.1-Clause: 0.829
enjoy (0.325) + *everything* (0.0)
F3.1-Predicate: 0.325
Clause-1 (0.829) + Clause-2 (0.325)
F5.1-Complex Clause: 0.884

Fig. 7. Sentiment analysis of an example sentence.

First of all, the contextual sentiment score of the object '*great movie*' is calculated using the formula F1.1 in Table 6. The prior sentiment scores of the noun '*movie*' and the adjective '*great'* are 0 and +0.625 respectively. Thus, the contextual sentiment score of the phrase '*great movie*' is calculated as + (0 + (1–0) * 0.625) which is equal to +0.625. Subsequently, the contextual sentiment score of the verb phrase '*watch happily*' is calculated by using the formula F2.1 in Table 7. The prior sentiment scores of the verb '*watch*' and the adverb '*happily*' are +0.089 and +0.5 respectively. Thus, the contextual score of the phrase '*watch happily*' is calculated as + (0.089 + (1–0.089) * 0.5) which is equal to +0.544. For the predicate '*happily watch this great movie*', sentiment score is calculated using the formula F3.1 in Table 8. The sentiment scores of the verb phrase and the object are +0.544 and +0.625 respectively. Thus, it is calculated as + (0.625 + (1–0.625) * 0.544) which is equal to +0.829. Then the score for the independent clause '*I can happily watch this great movie*' is calculated by using the formula F4.1 in Table 9. Since the prior score of the subject '*I*' is 0, the score of the clause remains +0.829. The contextual sentiment score of the predicate '*enjoy everything*' is also calculated based on the formula F3.1 in Table 8. Finally, the formula F5.1 in Table 10 is used to calculate the complex-to dependency. Thus, it is calculated as + (0.325 + (1–0.325) * 0.829) and the final contextual sentiment score becomes +0.884.

3.6. Review aspects

After calculating the contextual sentiment score for each clause in a sentence, the next step is to determine the review aspect of the clause using the algorithm shown in Figure 8. This process makes use of the semantic tags annotated by the semantic annotation process. If a semantic tag representing a specific aspect is found in a clause, it is categorized into the specific aspect. If no semantic tag is found in a clause and there is a previous or a following clause in the sentence, it is categorized into the same target aspect of the previous or following clause. Otherwise, it is categorized into the overall review aspect.

3.7. Sentence-level sentiment score

Although the aim of this study is to analyse the sentiment towards each aspect of a movie and calculate the clause-level sentiment score for each aspect, the sentence-level sentiment score is also calculated. A sentence can have one clause about one aspect, multiple clauses about the same aspect, or multiple clauses about various aspects. Therefore, after calculating the contextual sentiment scores at clause-level and determining their review aspects in a sentence, the sentiment score for each review aspect is calculated by grouping together the same aspect clauses and taking the average score. Then, the sentence-level sentiment score is calculated by averaging the sentiment scores for all aspects (i.e. both positive and negative scores). So, the sentiment score of the sentence '*this movie is good*' (one positive clause about overall aspect) is more positive than the sentiment score of the sentence '*the movie is good but I do not like the music*' (an additional negative clause about the music aspect).

```
FOR each clause
    IF [Music Semantic Tags] found THEN
        Review Aspect = Music;
    ELSE IF [Scene Semantic Tags] found THEN
        Review Aspect = Scene;
    ELSE IF [Story Semantic Tags] found THEN
        Review Aspect = Story;
    ELSE IF [Director Semantic Tags] found THEN
        Review Aspect = Director;
    ELSE IF [Cast Semantic Tags] found THEN
        Review Aspect = Cast;
    ELSE IF [Overall Semantic Tags] found THEN
        Review Aspect = Overall;
    ELSE IF previous or following clause exists THEN
        Review Aspect = Previous/Following Aspect;
    ELSE
        Review Aspect = Overall;
    END IF
END FOR
```

Fig. 8. Algorithm for review aspect.

4. Experiments

4.1. Datasets

We conducted experiments with a dataset of 1000 sentences: 500 positive and 500 negative. The movie review sentences were manually collected from the discussion board of a movie review site (www.imdb.com). For the experiments, our own dataset was used because aspect level sentiment labels are required to verify the effectiveness of our aspect-based sentiment analysis approach. Most of the publicly available movie review datasets contain only document level or sentence level sentiment labels.

For the dataset collection, firstly, 34 movies were selected: 17 positive and 17 negative movies based on the user ratings at the website. Then discussion threads in the selected movies were chosen randomly, and positive or negative sentences in the posts were selected manually, while posts with irrelevant and spam contents were ignored. When selecting sentences, we tried to collect a reasonable number of sentences (or clauses) for each aspect. The numbers of clauses for the review aspects are 583 for overall, 87 for director, 225 for cast, 127 for story, 104 for scene, and 27 for music.

Two coders read the collected sentences and manually classified sentiment orientations toward target aspects. Their decisions were compared to calculate intercoder reliability. We used Cohen's kappa coefficient to measure the agreement between the two independent coders who classified each clause in a sentence into positive, negative or neutral for six review aspects: *overall, director, cast, story, scene* and *music*. In our experiment, the intercoder agreement using Cohen's kappa coefficient was 0.93 which is considered to be a good agreement [47]. Conflicting labels by the two coders were reviewed and manually re-classified by one of the authors and these manually classified sentiment labels were used as answer keys.

Manual construction of the dataset was necessary to filter out several types of non-relevant sentences – opinion spam postings, reviews of other movies, non-subjective sentences, and so on. Automatic detection of these types of postings and sentences are non-trivial challenges, and not within the scope of this study. Opinion spam refers to postings that give undeserving positive reviews to some target product in order to promote the product, and/or give unjust or malicious negative reviews to other products in order to damage their reputation. For instance, Jindal and Liu [49] studied opinion spam in product reviews on Amazon.com, and built a logistic regression model for opinion spam detection. Their study suggests that more work is needed to identify important features for detecting opinion spam. We have not found opinion spam to be common in movie

reviews compared to product reviews. Another type of spam contains advertising material not relevant to the current movie, such as spam for an e-commerce website. Other non-relevant content includes sentences that discuss other movies instead of the current one, or other topics that are irrelevant to the current movie, such as the reviewer's favourite movie theatres. Also, many sentences are objective or factual sentences that do not carry any opinion. Automatic identification of objective versus subjective sentences has been studied separately by several researches [13, 37]. In sentiment analysis studies, it is common practice to filter out objective sentences and process only the remaining subjective sentences. For instance, Pang and Lee [17] applied sentiment categorization techniques only to the subjective portions of the document.

Admittedly, there may be some bias in the manual selection of sentences for the dataset. So we have constructed an additional dataset semi-automatically. The dataset construction was carried out more systematically to reduce bias. A computer program selected relevant discussion threads by examining the thread titles and the first post in the thread, and compared them against the feature term list of Table 2 and a set of sentiment bearing words in the domain specific and generic opinion lexicons. This yielded 511 sentences. Both automatic and manual selection of relevant sentences was performed in order to assess the accuracy of the automatic filtering of relevant sentences. However, evaluation of the sentiment analysis was performed with the manually filtered relevant sentences, comprising 342 sentences. Details are reported in Section 4.3. We first report our results with the dataset of 1000 sentences.

4.2. Experimental results with the dataset of 1000 sentences

Table 13 shows the precision, recall, and *F*-score for determining the review aspects of the clauses in the dataset of 1000 sentences. The approach makes use of the movie aspect and movie-specific feature terms in Table 2 for semantic annotation, and then the simple rules described in Figure 8 are applied to determine the review aspect of the clause. The results show that the simple rules are effective for review aspect determination. Precision, recall and *F*-score for each aspect are calculated using the following formulas:

$$\text{Precision} = \frac{\text{number of correctly tagged clauses}}{\text{number of automatically tagged clauses}} \tag{2}$$

$$\text{Recall} = \frac{\text{number of correctly tagged clauses}}{\text{number of manually tagged relevant clauses}} \tag{3}$$

$$F\text{-score} = 2 \times \frac{\text{precision} \times \text{recall}}{\text{precision} + \text{recall}} \tag{4}$$

Since the experiments are conducted with our own dataset, we cannot directly compare our results with other approaches. For comparison, we could simulate other approaches with our dataset, but it would require various resources used in the other approaches and these resources

Table 13
Precision, recall, and *F*-score for determining aspects

Aspect	Precision (%)	Recall (%)	*F*-score (%)
Overall	93.26	97.26	95.21
Direct	97.73	98.85	98.29
Cast	91.18	96.44	93.74
Story	95.16	92.91	94.02
Scene	95.33	98.08	96.68
Music	96.15	92.59	94.34

are not readily available. Therefore, we implement two baseline approaches in order to provide benchmarks for comparison with our approach. Tables 14 and 15 show precision, recall, *F*-score and accuracy of the baseline approaches by comparing the system results to the answer keys (gold standard). Precision, recall and accuracy are calculated using the following formulas (*F*-score is calculated with Equation (4)):

$$\text{Precision} = \frac{\text{number of correctly classified positive (or negative) sentences or clauses}}{\text{number of automatically classified positive (or negative) sentences or clauses}} \quad (5)$$

$$\text{Recall} = \frac{\text{number of correctly classified positive (or negative) sentences or clauses}}{\text{number of manually classified relevant positive (or negative) sentences or clauses}} \quad (6)$$

$$\text{Accuracy} = \frac{\text{number of correctly classified positive and negative sentences or clauses}}{\text{number of manually classified relevant positive and negative sentences or clauses}} \quad (7)$$

The baseline approaches do not use any syntactic parsing. This is to determine whether syntactic analysis used in our approach offers any performance advantages. Only the part-of-speech tags of lexical terms are used when matching the entries in the domain-specific and generic opinion lexicons to determine the terms' sentiment orientations and prior sentiment scores. For clause-level sentiment calculation, a clause for a particular aspect is extracted by locating the aspect's feature term (e.g. '*acting*' for cast aspect) and sentiment terms (e.g. '*good*') that co-occur in the sentence, without considering any grammatical dependencies. When a sentiment term is located in between two feature terms, the feature term within a shorter distance is selected with the sentiment term, and if they are within the same distance, the first feature term is selected. The first baseline approach (results shown in Table 14) uses a simple sentiment word count method – counting the numbers of positive and negative terms in each sentence or clause, and determining the sentiment orientation by comparing the counts of positive and negative terms. If the count of positive terms is larger (smaller) than that of negative terms, it is considered positive (negative). The second baseline approach (results shown in Table 15) uses the prior sentiment scores of the sentiment terms to consider their sentiment intensities (i.e. weights). Instead of counting positive and negative sentiment terms as in the first approach, it adds up the prior sentiment scores of positive and negative terms in each sentence or clause, and the sentiment orientation is determined by the total sentiment score. If the total score is positive (negative), it becomes positive (negative).

Table 14
Precision, recall, *F*-score, and accuracy of the baseline sentiment word count approach

		Precision (%)	Recall (%)	*F*-score (%)	Accuracy (%)
Sentence level	Positive	71.50	85.80	78.00	64.40
	Negative	88.11	43.00	57.80	(644/1000)
Clause level					
Overall aspect	Positive	62.30	80.17	70.11	56.09
	Negative	87.82	41.39	56.26	(327/583)
Direct aspect	Positive	84.62	77.19	80.73	57.47
	Negative	66.67	30.00	41.38	(50/87)
Cast aspect	Positive	80.81	83.33	82.05	44.44
	Negative	76.92	37.74	50.63	(100/225)
Story aspect	Positive	76.56	83.05	79.67	59.06
	Negative	96.30	44.83	61.18	(75/127)
Scene aspect	Positive	83.33	89.55	86.33	73.08
	Negative	94.12	50.00	65.31	(76/104)
Music aspect	Positive	86.67	76.47	81.25	59.26
	Negative	75.00	94.12	83.48	(16/27)
Clause level average	Positive	79.05	81.63	80.03	58.23
	Negative	82.80	49.68	59.71	

Table 15
Precision, recall, *F*-score, and accuracy of the baseline prior sentiment score approach

		Precision (%)	Recall (%)	*F*-score (%)	Accuracy (%)
Sentence level	Positive	74.29	83.80	78.76	74.00
	Negative	87.95	64.20	74.22	(740/1000)
Clause level					
Overall aspect	Positive	65.26	78.48	71.26	67.07
	Negative	88.74	61.93	72.95	(391/583)
Direct aspect	Positive	82.69	75.44	78.90	60.92
	Negative	62.50	50.00	55.56	(53/87)
Cast aspect	Positive	82.83	85.42	84.10	49.78
	Negative	90.91	56.60	69.77	(112/225)
Story aspect	Positive	72.58	76.27	74.38	59.06
	Negative	93.75	51.72	66.67	(75/127)
Scene aspect	Positive	86.76	88.06	87.41	75.96
	Negative	95.24	62.50	75.47	(79/104)
Music aspect	Positive	90.91	58.82	71.43	51.85
	Negative	66.67	95.24	78.43	(14/27)
Clause level average	Positive	80.17	77.08	77.91	60.77
	Negative	82.97	63.00	69.81	

Table 16 shows precision, recall, *F*-score and accuracy of the proposed sentiment analysis approach. It is observed that the proposed approach consistently yielded better results than both baseline approaches. We carried out a statistical test of difference in population proportion [48] to find out whether the percentage accuracy for the proposed approach (shown in Table 16) is significantly higher than for the prior sentiment score approach (shown in Table 15). Percentage accuracy is significantly higher for the proposed approach for the sentence level and all the clause level aspects.

In contrast to the baseline word count approach, the proposed approach also provides more quantitative sentiment analysis for comparing different clauses or sentences. For example, the proposed approach can determine that '*this movie is superb*' is more positive than the clause '*this movie is good*'. In the word count approach, they are assigned the same score since both sentences contain one positive sentiment term. In general, the word count approach has poor recall for negative sentences (see the recall values for negative clauses or sentences in Table 14). For example, the sentence '*this is a pretty bad story*' has one positive word and one negative word, and thus the word count will be equal and the result becomes neutral. Similarly, the sentence '*all good cast members are bad in this movie*' contains both '*good*' and '*bad*', and the approach fails to recognize the sentiment correctly. The baseline prior sentiment score approach shows better results than the word count approach, but it is not as effective as the proposed approach. Particularly when a clause or sentence contains terms with different sentiment orientations, the grammatical relationships between these terms should be considered in order to perform more accurate contextual sentiment analysis. This effort may not introduce significant improvement when document level sentiment analysis is applied. However, when sentence or clause level sentiment analysis is applied, such detailed linguistic analysis becomes important.

Tables 17 and 18 give examples of the most positive and negative sentences with their aspects and sentiment scores determined by the proposed approach.

4.3. Experimental results with the second dataset of 342 sentences

A second dataset was systematically constructed using a data crawler that filters irrelevant discussion threads by analysing thread titles and the first post in each thread. Only the threads which contain both feature terms (in Table 2) and sentiment words in the domain specific and generic opinion lexicons were selected for further analysis. The feature terms are generic terms (e.g. 'this movie' and 'the music') or movie specific metadata (e.g. movie, director and cast names). This

Table 16
Precision, recall, *F*-score, and accuracy of the proposed sentiment analysis method

		Precision (%)	Recall (%)	*F*-score (%)	Accuracy (%)
Sentence level	Positive	80.49	85.80	83.06	**82.30*** **
	Negative	85.53	78.00	81.59	(823/1000)
Clause level					
Overall aspect	Positive	71.81	78.48	75.00	75.13**
	Negative	84.80	76.06	80.19	(438/583)
Direct aspect	Positive	88.89	91.80	90.32	86.21***
	Negative	81.82	72.00	76.60	(75/87)
Cast aspect	Positive	87.69	88.37	88.03	82.67***
	Negative	84.71	81.82	83.24	(186/225)
Story aspect	Positive	79.17	96.61	87.02	79.53 ***
	Negative	97.78	74.58	84.62	(101/127)
Scene aspect	Positive	90.14	95.52	92.75	90.38 **
	Negative	93.33	80.00	86.15	(94/104)
Music aspect	Positive	94.12	88.89	91.43	81.48 *
	Negative	75.00	85.71	80.00	(22/27)
Clause level average	Positive	85.30	89.95	87.43	**82.57 *** **
	Negative	86.24	78.36	81.80	

*Significant at the 0.05 level; **significant at the 0.01 level; ***significant at the 0.001 level.

Table 17
Sentences with the most positive sentiment scores

Sentence	Aspect	Score
It features some fine acting, a terrific ending and it is deeply moving and touching at times	Cast	0.99
There's no question that it's the strongest of the franchise as a whole	Overall	0.98
This delightful little thriller is so fantastic	Overall	0.98
All the elements in this film gel perfectly together to make one superb masterpiece	Overall	0.98
This is also a wonderful tribute to Polish artists, through Chopin's music	Music	0.97
The special effects are excellently rendered, and really give their money's worth	Scene	0.97
Catherine Zeta Jones is gorgeous in this movie – a perfect casting choice	Cast	0.97

Table 18
Sentences with the most negative sentiment scores

Sentence	Aspect	Score
It was so bad in the first half hour, that my friend and I left	Overall	–1.00
It's so bad, it literally hurts	Overall	–1.00
It's basically a well made bad movie	Overall	–1.00
The writing in this film is horribly unrealistic and atrocious, so terrible it made me vomit	Story	–0.96
I didn't care for the music soundtrack. Seemed distracting and inappropriate	Music	–0.95
This is the first movie that I actually switched off after 20 min because it was that terrible	Overall	–0.94
Nail biting, edge of your seat, nerve wracking, stomach turning	Overall	–0.92
This movie to anyone who enjoys hilariously bad movies and can't wait to watch it	Overall	–0.91

yielded 511 sentences. After that, the filtering module identifies relevant sentences from each thread automatically by looking for the presence of sentiment terms. This simple method manages to eliminate objective sentences, but is not able to remove other types of non-relevant sentences, such as spam and off-topic sentences.

Manual filtering and annotation was then performed. Two coders tagged the sentences as relevant or non-relevant, and coded the sentiment orientation toward target aspects. The intercoder agreement using Cohen's kappa coefficient is 0.91 which is considered to be a good agreement. Conflicting labels by the two coders were reviewed and fixed by one of the authors, and these

manually classified sentiment labels were used as the answer keys. The manual tagging identified 342 sentences that are relevant to the current movies. The filtering module had earlier classified 441 sentences as relevant, of which 340 sentences are correct and 101 sentences wrong. This yielded a precision of 77% (340/441) and a recall of 99% (340/342). We analysed the misclassification errors and found that these sentences contain sentiment words, but discuss other topics such as other movies (e.g. 'I like the first one' and 'TDK is better') and off-topic arguments with other users (e.g. 'get a life' and 'what kind of movie do you like'). We observed that filtering non-relevant sentences is a challenging problem and should be studied separately in future work.

We carried out sentiment analysis only with the 342 relevant sentences. Among the non-relevant sentences, objective sentences do not contain sentiment expressions, and spam and off-topic sentences do not carry movie review aspects information. We assume that non-relevant sentences are removed before the sentiment analysis starts. Tables 19 and 20 show precision, recall, *F*-score and accuracy of the baseline prior sentiment score approach and the proposed sentiment analysis

Table 19
Precision, recall, *F*-score, and accuracy of the baseline prior sentiment score approach on the 342-sentence dataset

		Precision (%)	Recall (%)	*F*-score (%)	Accuracy (%)
Sentence level	Positive	69.11	78.57	73.54	67.54
	Negative	82.50	56.90	67.35	(231/342)
Clause level					
Overall aspect	Positive	57.45	76.42	65.59	62.55
	Negative	68.04	55.00	60.83	(147/235)
Direct aspect	Positive	71.43	55.56	62.50	53.85
	Negative	100.00	40.00	57.14	(7/13)
Cast aspect	Positive	78.79	66.67	72.22	58.02
	Negative	95.45	48.84	64.62	(47/81)
Story aspect	Positive	57.14	50.00	53.33	55.00
	Negative	87.50	58.33	70.00	(11/20)
Scene aspect	Positive	78.57	73.33	75.86	59.26
	Negative	83.33	41.67	55.56	(16/27)
Music aspect	Positive	50.00	50.00	50.00	62.50
	Negative	50.00	83.33	62.50	(5/8)
Clause level average	Positive	65.56	62.00	63.25	58.53
	Negative	80.72	54.53	61.77	

Table 20
Precision, recall, *F*-score, and accuracy of the proposed sentiment analysis method on the 342-sentence dataset

		Precision (%)	Recall (%)	*F*-score (%)	Accuracy (%)
Sentence level	Positive	82.82	82.82	82.82	**82.16*** **
	Negative	84.88	82.02	83.43	**(281/342)**
Clause level					
Overall aspect	Positive	76.58	79.44	77.98	80.00***
	Negative	81.75	82.40	82.07	(188/235)
Direct aspect	Positive	100.00	66.67	80.00	76.92
	Negative	80.00	80.00	80.00	(10/13)
Cast aspect	Positive	77.14	69.23	72.97	72.84*
	Negative	76.19	74.42	75.29	(59/81)
Story aspect	Positive	66.67	50.00	57.14	65.00
	Negative	90.00	75.00	81.82	(13/20)
Scene aspect	Positive	86.67	86.67	86.67	85.19*
	Negative	90.91	83.33	86.96	(23/27)
Music aspect	Positive	33.33	25.00	28.57	62.50
	Negative	60.00	90.91	72.29	(5/8)
Clause level average	Positive	73.40	62.83	67.22	**73.74*** **
	Negative	79.81	81.01	79.74	

*Significant at the 0.05 level; **significant at the 0.01 level; ***significant at the 0.001 level.

approach respectively. Since the discussion threads were harvested automatically, there are only a few sentences for some review aspects, such as director and music aspects. It is observed that the proposed approach consistently yielded better results than the baseline approach.

4.3.1. Error analysis

After carefully analysing the errors, the sources of errors are categorized into prior score, user error (spelling, grammar, capitalization, and so on), indirect expression (misleading terms), and algorithm (parser, rules, negation, and so on) as shown in Figure 9.

Some of the errors come from the wrong assignment of prior scores to the words. As shown in the following example, the word '*satisfied*' is a negative word in SentiWordNet although it should be considered positive. Comprehensive re-entry of these words to the domain specific lexicons is not feasible and, as a result, some clauses are wrongly classified:

- *I really felt satisfied* (Answer Key = Positive; System Result = Negative*).

Some of the words were given wrong prior scores due to their multiple senses with different sentiment values. As shown in the following examples, the word '*miss*' has different senses. In our approach, the prior score of '*miss*' is negative, and the output becomes positive because of the negation. Thus, the second example is wrongly classified:

- *You should not miss this movie* (Answer Key = Positive; System Result = Positive);
- *Bye, we will not miss you!* (Answer Key = Negative; System Result = Positive*).

To overcome this issue, we may detect the right sense of the word '*miss*' in context using a word sense disambiguation algorithm. However, word sense disambiguation itself is a challenging problem and error prone, so it is not commonly used in sentiment analysis.

Some of the clauses express sentiment indirectly and contain misleading terms which caused the system to classify them incorrectly:

- *Well deserved 1/10* (Answer Key = Negative; System Result = Positive*);
- *Packed a tremendous punch* (Answer Key = Positive; System Result = Negative*).

Users often make typographical errors or grammatical mistakes. The system corrects capitalization errors automatically, but not the spellings. When spellings are wrong, the prior scores are not assigned properly:

- *The story was so borning!* (Answer Key = Negative; System Result = Positive*).

The majority of misclassifications are made by the algorithm because it cannot handle some complex expressions of sentiment in texts. Grammatical parsing is sometimes inaccurate for lengthy complex sentences, and thus it constructs wrong dependency trees. In some cases the system fails to handle the negation of clauses correctly as shown in the examples below:

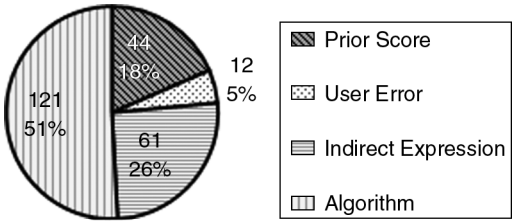

Fig. 9. Distribution of error sources.

• *Nothing will ever suck as much* (Answer Key = Negative; System Result = Positive*);
• *Couldn't believe my eyes at this pathetic movie* (Answer Key = Negative; System Result = Positive*).

Incomplete sentences are often parsed incorrectly. As shown in the example below, the term *'blunt'* is wrongly tagged as verb. With wrong POS information, the prior score cannot be assigned correctly and thus, the clause is wrongly classified. If the word was tagged correctly as adjective, the prior score could be assigned correctly:

• *Extraordinarily blunt filmmaking* (Answer Key = Negative; System Result = Positive*).

5. Conclusion

Sentiment analysis of review documents should consider multiple sentiments towards different aspects of the reviewed entity. The proposed approach performs sentiment analysis at the clause level so that sentiments for different aspects can be analysed separately. The system processes the grammatical dependencies of words in a sentence, divides it into independent clauses and calculates the contextual sentiment score of each clause focusing on a specific aspect. The experimental results show that the proposed approach is effective for aspect-based sentiment analysis of short documents such as message posts on discussion boards. The accuracies of clause level sentiment classification for overall movie, director, cast, story, scene and music aspects are 75%, 86%, 83%, 80%, 90% and 81% respectively.

This approach can be used for sentiment summarization of multiple review aspects. Unlike other approaches that focused on extracting feature–opinion pairs, the proposed approach provides the sentiment score of a clause or sentence, and it allows comparison of the clause or sentence with others in terms of sentiment strength. For instance, the sentiment scores provided by the proposed approach can be used for highlighting the most positive and negative clauses or sentences associated with particular aspects. Our future work will study sentiment summarization across multiple genres (such as forums, user reviews, critic reviews and Twitter) which will allow the readers to search and browse movies based on system-calculated sentiment scores, aspects, user-provided sentiment ratings or an individual reader's preference (e.g. a sentiment summary only from critic reviews).

References

[1] C.-M. Chiu, Towards a hypermedia-enabled and web-based data analysis framework, *Journal of Information Science* 30(1) (2004) 60–72.
[2] J. Wang (ed), *Encyclopedia of Data Warehousing and Mining* (Information Science Reference, Hershey, 2008).
[3] R. Quirk, S. Greenbaum, G. Leech and J. Svartvik, *A Comprehensive Grammar of the English Language* (Longman, London, 1985).
[4] P. Chaovalit and L. Zhou, Movie review mining: a comparison between supervised and unsupervised classification approaches, *Proceedings of the 38th Annual Hawaii International Conference on System Sciences* (2005).
[5] P.D. Turney, Thumbs up or thumbs down? Semantic orientation applied to unsupervised classification of reviews, *Proceedings of the 40th Annual Meeting of the Association for Computational Linguistics ACL* (2002) 417–434.
[6] A. Esuli and F. Sebastiani, Determining term subjectivity and term orientation for opinion mining, *Proceedings of the European Chapter of the Association for Computational Linguistics* (2006) 193–200.
[7] T. Wilson, J. Wiebe and P. Hoffmann, Recognizing contextual polarity in phrase-level sentiment analysis, *Proceedings of the Conference on Human Language Technology and Empirical Methods in Natural Language Processing* (2005) 347–354.
[8] J. Wiebe, Tracking point of view in narrative, *Computational Linguistics* 20(2) (1994) 233–287.

[9] P.J. Stone, D.C. Dunphy, M.S. Smith and D.M. Ogilvie, *General Inquirer: a Computer Approach to Content Analysis* (The MIT Press, Cambridge, MA, 1966).
[10] V. Hatzivassiloglou and K.R. McKeown, Predicting the semantic orientation of adjectives, *Proceedings of the 35th Annual Meeting of the ACL and the 8th Conference of the European Chapter of the ACL* (1997) 174–181.
[11] J. Wiebe, Learning subjective adjectives from corpora, *Proceedings of the 17th National Conference on Artificial Intelligence* (2000) 735–740.
[12] E. Riloff, J. Wiebe and T. Wilson, Learning subjective nouns using extraction pattern bootstrapping, *Proceeding of the 7th Conference on Natural Language Learning* (2003) 25–32.
[13] E. Riloff and J. Wiebe, Learning extraction patterns for subjective expressions, *Proceedings of the 2003 Conference on Empirical Methods in Natural Language Processing* (2003) 105–112.
[14] G. Qiu, B. Liu, J. Bu and C. Chen, Expanding domain sentiment lexicon through double propagation, *Proceedings of the 21st International Joint Conference on Artificial Intelligence* (Morgan Kaufmann, San Francisco, 2009) 1199–1204.
[15] K. Dave, S. Lawrence and D.M. Pennock, Mining the peanut gallery: opinion extraction and semantic classification of product reviews, *Proceedings of the 12th International Conference on World Wide Web* (2003) 519–528.
[16] T. Mullen and N. Collier, Sentiment analysis using support vector machines with diverse information sources, *Proceedings of the 9th Conference on Empirical Methods in Natural Language Processing* (2004) 412–418.
[17] B. Pang and L. Lee, Seeing stars: exploiting class relationships for sentiment categorization with respect to rating scales, *Proceedings of the Association for Computational Linguistics* (2005) 115–124.
[18] B. Pang, L. Lee and S. Vaithyanathan, Thumbs up? Sentiment classification using machine-learning techniques, *Proceedings of the 2002 Conference on Empirical Methods in Natural Language Processing* (2002) 79–86.
[19] C. Whitelaw, N. Garg and S. Argamon, Using appraisal groups for sentiment analysis, *Proceedings of the 14th ACM Conference on Information and Knowledge Management* (2005) 625–631.
[20] J.-C. Na and T.T. Thet, Effectiveness of web search results for genre and sentiment classification, *Journal of Information Science* 35(6) (2009) 709–727.
[21] L. Zhou and P. Chaovalit, Ontology-supported polarity mining, *Journal of the American Society for Information Science and Technology* 59(1) (2008), 98–110.
[22] L.-W. Ku and H.-H. Chen, Mining opinions from the Web: beyond relevance retrieval, *Journal of the American Society for Information Science and Technology* 58(12) (2007), 1838–1850.
[23] T. Joachims, Text categorization with support vector machines: learning with many relevant features, *Proceedings of the 10th European Conference on Machine-learning* (1998) 137–142.
[24] J.R. Martin and P.R.R. White, *The Language of Evaluation: Appraisal in English* (Palgrave Macmillan, London, 2005).
[25] J.-C. Na, T.T. Thet and C.S.G. Khoo, Comparing sentiment expression in movie reviews from four online genres, *Online Information Review* 34(2) (2010) 317–338.
[26] S. Blair-Goldensohn, K. Hannan, R. McDonald, T. Neylon, G.A. Reis and J. Reynar, Building a sentiment summarizer for local service reviews, *Proceedings of WWW 2008 Workshop: NLP Challenges in the Information Explosion Era* (2008).
[27] X. Ding, B. Liu and P.S. Yu, A holistic lexicon-based approach to opinion mining, *Proceedings of the International Conference on Web Search and Web Data Mining* (ACM, New York, 2008) 231–240.
[28] M. Hu and B. Liu, Mining and summarizing customer reviews, *Proceedings of the 10th ACM SIGKDD International Conference on Knowledge Discovery and Data Mining* (2004) 168–177.
[29] M. Ganapathibhotla and B. Liu, Mining opinions in comparative sentences, *Proceedings of the International Conference on Computational Linguistics* (2008) 241–248.
[30] S.-M. Kim and E. Hovy, Automatic detection of opinion bearing words and sentences, *Proceedings of the International Joint Conference on Natural Language Processing* (2005) 61–66.
[31] T. Miyoshi and Y. Nakagami, Sentiment classification of customer reviews on electric products, *Proceedings of International Conference on Systems, Man and Cybernetics* (2007) 2028–2033.
[32] M.A.M Shaikh, H. Prendinger and M. Ishizuka, Sentiment assessment of text by analyzing linguistic features and contextual valence assignment, *Applied Artificial Intelligence* 22(6) (2008) 558–601.
[33] J. Yi, T. Nasukawa, R. Bunescu and W. Niblack, Sentiment analyzer: extracting sentiments about a given topic using natural language processing techniques, *Proceedings of the 3rd IEEE International Conference on Data Mining* (2003) 427–434.

[34] C. Zhang, D. Zeng, J. Li, F.-Y. Wang and W. Zuo, Sentiment analysis of Chinese documents: from sentence to document level, *Journal of the American Society for Information Science and Technology* 60(12) (2009) 2474–2487.
[35] O. Vechtomova, Facet-based opinion retrieval from blogs, *Information Processing and Management* 46(1) (2010) 71–88.
[36] A.L. Berger, V.J. Della Pietra and S.A. Della Pietra, A maximum entropy approach to natural language processing, *Computational Linguistics* 22(1) (1996) 39–71.
[37] J. Wiebe and E. Riloff, Creating subjective and objective sentence classifiers from unannotated texts, *Proceedings of the International Conference on Intelligent Text Processing and Computational Linguistics* (2005) 475–486.
[38] T. Wilson, J. Wiebe and R. Hwa, Recognizing strong and weak opinion clauses, *Computational Intelligence* 22(2) (2006) 73–99.
[39] R.E. Schapire and Y. Singer, BoosTexter: A boosting-based system for text categorization, *Machine Learning* 39(2/3) (2000) 135–168.
[40] W. Cohen, Learning trees and rules with set-valued features, *Proceedings of the 13th National Conference on Artificial Intelligence/8th Innovative Applications of Artificial Intelligence Conference (AAAI/IAAI) Volume 1* (1996) 709–716.
[41] K. Toutanova and C.D. Manning, Enriching the knowledge sources used in a maximum entropy part-of-speech tagger, *Proceedings of the Joint SIGDAT Conference on Empirical Methods in Natural Language Processing and Very Large Corpora (EMNLP/VLC-2000)* (2000) 63–70.
[42] T.M. Mitchell, *Machine Learning* (McGraw Hill, New York, 1997).
[43] S.E. Robertson and K. Sparck Jones, Relevance weighting of search terms, *Journal of the American Society for Information Science* 27(3) (1976) 129–146.
[44] Y. Yang and J.O. Pedersen, A comparative study on feature selection in text categorization, *Proceedings of the Fourteenth International Conference on Machine Learning (ICML '97)* (1997) 412–420.
[45] M.-C. de Marneffe, B. MacCartney and C.D. Manning, Generating typed dependency parses from phrase structure parses, *Proceedings of the 5th International Conference on Language Resources and Evaluation* (2006).
[46] M.-C. de Marneffe and C.D. Manning, *Stanford typed dependencies manual* (2010). Available at: http://nlp.standford.edu/software/dependencies_manual.pdf (accessed 10 October 2010).
[47] T. Byrt, How good is that agreement? *Epidemiology* 7(5) (1996) 561.
[48] J. Neter, W. Wasserman and G.A. Whitmore, *Applied Statistics* (Allyn and Bacon, Boston, MA, 1992).
[49] N. Jindal and B. Liu, Opinion Spam and Analysis, *Proceedings of the International Conference on Web Search and Web Data Mining (WSDM'08)* (Palo Alto, California, USA, 2008) 219–229.