

# Prior and contextual emotion of words in sentential context<sup>☆</sup>

Diman Ghazi<sup>a</sup>, Diana Inkpen<sup>a</sup>, Stan Szpakowicz<sup>a,b,\*</sup>

<sup>a</sup> School of Electrical Engineering and Computer Science, University of Ottawa, Ottawa, Canada

<sup>b</sup> Institute of Computer Science, Polish Academy of Sciences, Warsaw, Poland

Received 2 August 2012; received in revised form 4 April 2013; accepted 24 April 2013

Available online 10 May 2013

## Abstract

A set of words labeled with their prior emotion is an obvious place to start on the automatic discovery of the emotion of a sentence, but it is clear that context must also be considered. It may be that no simple function of the labels on the individual words captures the overall emotion of the sentence; words are interrelated and they mutually influence their affect-related interpretation. It happens quite often that a word which invokes emotion appears in a neutral sentence, or that a sentence with no emotional word carries an emotion. This could also happen among different emotion classes. The goal of this work is to distinguish automatically between prior and contextual emotion, with a focus on exploring features important in this task. We present a set of features which enable us to take the contextual emotion of a word and the syntactic structure of the sentence into account to put sentences into emotion classes. The evaluation includes assessing the performance of different feature sets across multiple classification methods. We show the features and a promising learning method which significantly outperforms two reasonable baselines. We group our features by the similarity of their nature. That is why another facet of our evaluation is to consider each group of the features separately and investigate how well they contribute to the result. The experiments show that all features contribute to the result, but it is the combination of all the features that gives the best performance.

© 2013 Elsevier Ltd. All rights reserved.

**Keywords:** Sentiment analysis; Polarity; Emotion; Prior emotion; Contextual emotion; Syntactic features; Natural language processing; Machine learning

## 1. Introduction and related work

Recognition, interpretation and representation of affect have been investigated by researchers in the field of affective computing (Picard, 1997). They have considered a wide range of modalities such as affect in speech, facial display, posture and physiological activity. It is only recently that there has been a growing interest in automatic identification and extraction of sentiment, opinions and emotions in *text*.

Sentiment analysis and opinion mining constitute the field of study which analyzes people's opinions, sentiments, evaluations, attitudes and emotions occurring in written language. In the early 2000s sentiment analysis has become one of the most active research areas in natural language processing (NLP), and a major application of data mining, Web mining and text mining. This wave of activity is due to the rapid growth of the social media – in particular reviews,

<sup>☆</sup> This paper has been recommended for acceptance by R.K. Moore.

\* Corresponding author. Tel.: +1 613 562 5800x6704

E-mail addresses: [dghaz038@uottawa.ca](mailto:dghaz038@uottawa.ca) (D. Ghazi), [Diana.Inkpen@uottawa.ca](mailto:Diana.Inkpen@uottawa.ca) (D. Inkpen), [szpak@eecs.uottawa.ca](mailto:szpak@eecs.uottawa.ca) (S. Szpakowicz).

discussion fora, blogs, micro-blogs, Twitter and social networks – and the ensuing easy access to a mass of subjective and emotional data recorded in the digital form.

Another reason why sentiment analysis is such an active area of NLP research is that its applications have spread to multiple domains, from consumer product reviews, health care and financial services to social events and political elections (Liu, 2012). Here are some of the best known applications: classifying positive and negative movie reviews (Pang et al., 2002; Turney, 2002), opinion question-answering (Yu and Hatzivassiloglou, 2003; Stoyanov et al., 2005), and summarizing customer reviews (Hu and Li, 2004).

Early research has mainly focused on determining the presence of sentiment in the given text, and on determining its polarity – the positive or negative orientation, a finding which is useful but insufficient in most real-life applications. Practicality often dictates a more in-depth and fine-grained analysis. The analysis of sentiment must therefore go beyond differentiating positive from negative emotions toward giving a systematic account of the qualitative differences among individual emotions (Ortony et al., 1988).

Emotions are among of the most pervasive aspects of human experience. They have been studied in multiple fields, psychology, philosophy, sociology among them. Recently, emotions have attracted the attention of researchers in computer science and particularly in computational linguistics. Computational methods of emotion analysis have also focused on various emotion modalities, but there was only limited effort in the direction of automatic recognition of emotion in text (Alm et al., 2005; Aman and Szpakowicz, 2007; Neviarouskaya et al., 2011).

In this work, we deal with assigning fine-grained emotion classes to sentences in text. On the face of it, our task is strongly related to polarity analysis, but categorization into distinct emotion classes is more difficult. That is not only because emotion recognition in general requires deeper insights but also because there are similarities between emotions which make clean classification a challenge. Particularly notable in this regard are anger and disgust, two emotion classes which even human annotators often find hard to distinguish (Aman and Szpakowicz, 2007).

A number of theories have been proposed, each with its own set of basic emotions, though with significant overlap. Ekman (1992) argued for six basic emotions with universally accepted distinctive facial expressions: joy, sadness, anger, fear, disgust and surprise. Plutchik (1962) added trust and anticipation to Ekman's set. Izard (1971) defined ten basic emotions: anger, contempt, disgust, distress, fear, guilt, interest, joy, shame and surprise. More recently, Parrot (2002) proposed six primary emotions: love, joy, surprise, anger, sadness and fear.

We categorize sentences into six basic emotions defined in Ekman (1992), a set most frequently adopted by NLP researchers. There also may, naturally, be no emotion in a sentence; that is, tagged as neutral/non-emotional.

Sentiment can be sought in text units at three levels (Wilson et al., 2009). While much work in sentiment analysis has targeted documents, more has been done with sentences. The third level of sentiment analysis is phrase-level, which, to the best of our knowledge, has not been used in emotion recognition research due to its complication and lack of data.

Document-level analysis assumes that each document expresses a single sentiment. While it is not unlikely for a document to express a single polarity, it is quite rare to find a document with only one emotion. On the other hand, one emotion is quite feasible in a sentence, though it is not uncommon for more than one emotion to be conveyed. For example, the sentence “Although the service is not that great, I still love this restaurant.” has a positive tone, but is not entirely positive. In such cases, we would have to classify emotions at the phrase level and allow sentences with multiple emotions. To train a classifier, we would need a dataset with mixed emotions marked. To the best of our knowledge, however, no such dataset is available yet. That is why we work with datasets from which sentences with conflicting emotion annotations have been removed, so we can assume that each sentence only indicates one emotion.

In order to recognize and analyze affect in written text – seldom explicitly marked for emotions – NLP researchers have come up with a variety of techniques, including machine learning, rule-based methods and the lexical method (Neviarouskaya et al., 2011).

The existence of an annotated corpus with rich information about opinions and emotions would support the development and evaluation of NLP systems which exploit such information. In particular, statistical NLP and machine learning have become the methods of choice for constructing a wide variety of practical NLP applications (Wiebe et al., 2005). Such methods have been previously applied to corpus-based features, mainly unigrams, combined with lexical features (Pang et al., 2002; Alm et al., 2005; Aman and Szpakowicz, 2007; Katz et al., 2007). The study by Pang et al. (2002) was among the first to describe a classification of movie reviews into positive and negative. The authors showed that using unigrams as features in classification performed quite well with Naive Bayes and with Support Vector Machines (SVM). In general, these two supervised machine learning algorithms have long been the

methods of choice for sentiment recognition in text. SVM has been shown to outperform Naive Bayes consistently in our particular task (Pang et al., 2002; Alm et al., 2005; Aman and Szpakowicz, 2007; Ghazi et al., 2010). The weakness of such methods is that they disregard negation, syntactic relations and semantic dependencies. They also require large annotated corpora for meaningful statistics and good performance. Processing may take time, and annotation effort is inevitably high.

Rule-based methods (Chaumartin, 2007; Neviarouskaya et al., 2011) have not been as popular as machine learning. They require a substantial manual rule-creation effort, an expensive process with weak guarantee of consistency and coverage, and likely very task-dependent. The set of rules for an affect analysis task (Neviarouskaya et al., 2011) can differ drastically from what underlies other tasks such as rule-based part-of-speech tagger, discourse parsers, word sense disambiguation and machine translation.

Another common take on emotion analysis is to use a lexicon with information about which words and phrases carry which emotion. The study of emotions in lexical semantics was the theme of a SemEval 2007 task (Strapparava and Mihalcea, 2007), carried out in an unsupervised setting (Strapparava and Mihalcea, 2008; Chaumartin, 2007; Kozareva, 2007; Katz et al., 2007). The participants were encouraged to work with *WordNet-Affect* (Strapparava and Valitutti, 2004) and *SentiWordNet* (Esuli and Sebastiani, 2006) to classify news headline sentences into six emotions. Word-level analysis, however, will not suffice when affect is expressed by phrases which require complex phrase- and sentence-level analyses: words are interrelated and they mutually influence their affect-related interpretation. On the other hand, words can have more than one sense, and they can only be disambiguated in context. Let us highlight several issues.

- An emotional word may have different orientation in different application domains (Fahrni and Klenner, 2008) – like the word *cool* when used to describe a car or someone’s demeanor.
- A sentence containing an emotional word may not express any emotion. Consider the vacuous polite *afraid* in “I’m afraid it’s going to rain.”
- Many sentences without emotional words can also convey emotions. Consider the rather negative opinion in “This washer uses a lot of water.”

As a result, the emotion conveyed by a word in a sentence can differ drastically from the emotion of the word on its own. We refer to the emotion listed for a word in an emotion lexicon as the word’s *prior* emotion. A word’s *contextual* emotion is the emotion of the sentence in which that word appears, when we take the context into account.

In this work, we focus on disambiguating the contextual emotion of words, taking a sentence as the context. Our method combines several ways of tackling the problem. First, we find keywords listed in the lexicon we use, and select the sentences which include emotional words from that lexicon. Next, we study the syntactic structure and semantic relations in the text surrounding the emotional word. We explore features important in emotion recognition, and we consider their effect on the emotion expressed by the sentence. Finally, we use machine learning to classify the sentence, represented by the chosen features, based on the contextual emotion of emotional words which occur in that sentence.

We evaluate our results by comparing our method applied to our set of features with Support Vector Machine (SVM) applied to *Bag-of-Words*, which was found to give the best performance among supervised methods (Yang and Li, 1999; Pang et al., 2002; Aman and Szpakowicz, 2007; Ghazi et al., 2010). We show that our method is promising and that it outperforms both a system which works only with prior emotions of words, ignoring context, and a system which applies SVM to *Bag-of-Words*. We also evaluate the usefulness of our features for this task by comparing the performance of each group of features with the baseline. We find that all the features help the performance of the classifier somewhat, and that certain features – such as part-of-speech and negation – improve the results significantly.

The remainder of the paper is structured as follows. Section 2 describes the dataset and the lexicons used in our experiments. Section 3 discusses the features which we find useful for recognizing contextual emotions. Experiments and results are presented in Section 4. In Section 5, we sum up our findings and discuss future work.

## 2. Dataset and resources

Supervised statistical methods typically require training data and test data, manually annotated with respect to each language-processing task to be learned. Due to the contributions of many researchers, several subjectivity and emotion

Table 1  
Distribution of labels in the *WordNet-Affect* lexicon.

happiness	sadness	anger	disgust	surprise	fear	total
398	201	252	53	71	141	1116

Table 2  
Distribution of labels in the prior-polarity lexicon.

Neutral	Negative	Positive	Both
6.9%	59.7%	31.1%	0.3%

Table 3  
Distribution of labels in the NRC-Emotion lexicon.

joy	sadness	anger	disgust	surprise	fear	trust	anticipation	pos.	neg.
689	1191	1247	1058	534	1476	1231	839	2312	3324

datasets and lexicons have been constructed, most of them publicly available. In this section, we explain the dataset and lexicons used in our experiments.

## 2.1. Lexicons

**WordNet-Affect.** The first resource we require is an emotional lexicon, a set of words which indicate the presence of a particular emotion. *WordNet-Affect* (Strapparava and Valitutti, 2004)<sup>1</sup> is organized exactly as its parent, *WordNet*: into a synset hierarchy by hypernymy/hyponymy. Affective labels are simply attached to synsets in that hierarchy. There is information in this resource – additional labels for positive, negative, neutral or ambiguous affect – which we did not use in this work. *WordNet-Affect* contains six lists of words corresponding to the six basic emotion categories. Table 1 shows their distribution.

**The prior-polarity lexicon.** The prior-polarity subjectivity lexicon (Wilson et al., 2009) contains over 8000 subjectivity clues collected from a number of sources. To create this lexicon, the authors began with the list of subjectivity clues extracted by Riloff (2003). The list was expanded using a dictionary and a thesaurus, and adding positive and negative word lists from the General Inquirer ([www.wjh.harvard.edu/inquirer/](http://www.wjh.harvard.edu/inquirer/)). Words are grouped into strong subjective and weak subjective clues; Table 2 presents the distribution of their polarity.

**The NRC-Emotion lexicon.** This lexicon (Mohammad and Turney, 2010) is a list of words and their associations with eight emotions (anger, fear, anticipation, trust, surprise, sadness, joy, and disgust) and two sentiments (negative and positive); this dataset has recently become publicly available. The manual annotation was constructed by crowd-sourcing, using Amazon’s Mechanical Turk. Each word-sense pair had at least three annotators, and the lexicon was created by taking the union of emotions associated with all the senses of a word. Table 3 displays the distribution of the emotion and polarity of words. In our experiments, we only consider Ekman’s six emotions, so we remove the trust and anticipation emotional words from the lexicon. The main difference between this lexicon and *WordNet-Affect* is that each word in the lexicon can belong to multiple emotion classes as well as both sentiment classes. As a result, while there are 6195 emotion words and their emotion class pairs for six emotion classes in the lexicon, it only has 3462 emotion word types. We do not use the positive and negative words, because the prior-polarity lexicon is larger and covers more words.

**The intensifier lexicon.** It is a list of 112 modifiers (adverbs) (Neviarouskaya et al., 2010). Two annotators gave coefficients for intensity degree – strengthening or weakening, from 0.0 to 2.0 – and the result was averaged.

<sup>1</sup> It was among the resources at [www.cse.unt.edu/rada/affectivetext/](http://www.cse.unt.edu/rada/affectivetext/) provided for SemEval 2007 Task 14.

Table 4

Distribution of labels in Aman’s modified dataset; the labels are happiness, sadness, anger, disgust, surprise, fear, and no emotion.

hp	sd	ag	dg	sr	fr	ne	total
536	173	179	172	115	115	800	2090

Table 5

Distribution of labels in the portions of Aman’s dataset used in our experiments, which we call part 1 (one emotional word), part 2 (more emotional words), and part 1 + part 2; the labels are happiness, sadness, anger, disgust, surprise, fear, and no emotion.

	hp	sd	ag	dg	sr	fr	ne	total
Part 1	196	64	64	63	36	52	150	625
Part 2	51	18	22	18	9	14	26	158
Part 1 + part 2	247	82	86	81	45	66	176	783

## 2.2. Training and test data

**Emotion dataset.** The main consideration in the selection of data for emotional classification task is that the data should be rich in emotion expressions. That is why we chose for our experiments a corpus of blog sentences annotated with emotion labels, discussed by Aman and Szpakowicz (2007). Each sentence is tagged by its dominant emotion, or as non-emotional if it does not include any emotion. The annotation is based on Ekman’s six emotions at the sentence level. The dataset contains 4090 annotated sentences, 68% of which were marked as non-emotional. The highly unbalanced dataset with non-emotional sentences as by far the largest class, and merely 3% in the fear and surprise classes, prompted us to remove randomly 2000 of the non-emotional sentences. This brought the number of non-emotional sentences down to 38% of all the sentences, and thus reduced the imbalance. Table 4 shows the details of the chosen dataset.

For our particular task we cannot even use the whole dataset. Our main consideration is to classify a sentence based on the contextual emotion of the words (known as emotional in the lexicon). The two emotion lexicons described in Section 2.1 serve this purpose; we use them separately to choose only the sentences which contain at least one emotional word according to the lexicon.

In the *NRC-Emotion* lexicon, however, each word can belong to multiple emotion classes. This actually happens quite often among negative emotion classes. For example, the word “hate” belongs to four negative emotion classes: anger, disgust, fear and sadness. As a result, if we choose sentences which contain at least one emotional word according to the *NRC-Emotion* lexicon, we will have 1010 sentences. Only 361 of those sentences will have one emotional word from exactly one emotion class (201 of them are *Happy* words). Part of our experiments are based on sentences with only one emotional word, so this makes the *NRC-Emotion* lexicon not suitable for our task, particularly for negative emotion classes. Even so, we use it for comparison purposes and to evaluate our features.

In effect, to prepare the dataset for our task, we use the *WordNet-Affect* lexicon. In the dataset, we only choose sentences which contain at least one emotional word according to *WordNet-Affect*. This gives us 783 sentences, 625 of which contain only one emotional word. The details appear in Table 5.

## 3. Features

As is any other application of supervised machine learning, the key in sentiment classification is the engineering of a set of effective features. Terms and their frequency are the most common features in traditional text classification, and they have been shown highly effective for sentiment classification as well (Liu, 2012). They cannot, however, detect relations and semantic dependencies. Those most common features serve only for comparison purposes.

The features used in our experiments were motivated both by the literature (Wilson et al., 2009; Choi et al., 2005) and by the exploration of contextual emotion of words in the annotated data. The counts of the feature values were based on the occurrences in the sentence of emotional words from the lexicons. For ease of description, we group the features into four distinct sets: emotion-word features, part-of-speech (POS) features, sentence features and dependency-tree

features. We will describe some of the features as multi-valued or categorical features. In practice, however, all features have been binarized.

**Emotion-word features.** This set of features is based on the emotion words themselves. Although the prior emotion and polarity of the emotional word in the sentence are insufficient for emotion analysis, they *are* useful for simple sentences such as “I am happy”, where the prior and contextual emotion of the sentence are exactly the same and there are no emotion shifters and influencers. These features have also been widely adopted and found useful in detecting sentiment and emotions:

- (a) the emotion of a word according to the emotion lexicon;
- (b) the polarity of a word according to the prior-polarity lexicon;
- (c) the presence of a word on a small list of modifiers.

**POS features.** Words in certain grammatical classes have been shown to be more effective in recognizing the emotion of a sentence. For example, adjectives and adverbs are important indicators of emotions (Mohammad and Turney, 2010). We use the Stanford tagger (Toutanova et al., 2003), which gives every word in a sentence a Penn Treebank tag.

The *WordNet-Affect* emotion lexicon lists the POS of the words along with their corresponding emotion. That is why we use the POS of the emotional word itself, both according to the emotion lexicon and to the Stanford tagger.

The POS of neighboring words in the same sentence are an important factor. We chose a  $[-2, +2]$  window, as suggested by the literature (Choi et al., 2005).

**Sentence features.** At present, we only have one feature in this group: the number of words in the sentence.

**Dependency-tree features.** These features capture the various types of relationships which involve the emotional word. The feature values come from a dependency parse tree of the sentence, obtained by parsing the sentence and then converting it to dependencies. We work with the Stanford parser (Marneffe et al., 2006). The dependencies are all binary relations: a grammatical relation holds between a governor (head) and a dependent (modifier). Fig. 1 shows an example.

According to Mohammad and Turney (2010), adverbs and adjectives are some of the most emotion-inspiring terms.<sup>2</sup> That is why we have decided to consider only a handful of the 52 different dependencies. In order to keep the number of features small, we only chose the negation, adverb and adjective modifier dependencies.

After parsing the sentence and getting the dependencies, we count three dependency-tree Boolean features for the emotional word.

- (a) If the word is in a “neg” dependency (negation modifier): true when there is a negation word which modifies the emotional word.
- (b) If the word is in a “amod” dependency (adjectival modifier): true when the emotional word is (i) a noun modified by an adjective or (ii) an adjective modifying a noun.
- (c) If the word is in a “advmod” dependency (adverbial modifier): true when the emotional word (i) is a non-clausal adverb or the head of an adverbial phrase which serves to modify the meaning of a word or (ii) has been modified by an adverb.

We also have several modification features based on the dependency tree. These Boolean features capture different types of relationships involving the emotional word. We list the feature name and the condition on the emotional word  $w$  which makes the feature true.

- (i) Modifies-positive:  $w$  modifies a positive word from the prior-polarity lexicon.
- (ii) Modifies-negative:  $w$  modifies a negative word from the prior-polarity lexicon.
- (iii) Modified-by-positive:  $w$  is the head of the dependency, which is modified by a positive word from the prior-polarity lexicon.

<sup>2</sup> This is not surprising given that adjectives qualify nouns and adverbs qualify verbs.

**Your query**

*It was the best summer I have ever experienced.*

**Tagging**

It/PRP was/VBD the/DT best/JJS summer/NN  
I/PRP have/VBP ever/RB experienced/VBN ./.

**Parse**

```
(ROOT
  (S
    (NP (PRP It))
    (VP (VBD was)
      (NP
        (NP (DT the) (JJS best) (NN summer))
        (SBAR
          (S
            (NP (PRP I))
            (VP (VBP have)
              (ADVP (RB ever))
              (VP (VBN experienced))))))
      (. .)))
```

**Typed dependencies**

```
nsubj(summer-5, It-1)
cop(summer-5, was-2)
det(summer-5, the-3)
amod(summer-5, best-4)
root(ROOT-0, summer-5)
nsubj(experienced-9, I-6)
aux(experienced-9, have-7)
advmod(experienced-9, ever-8)
rcmod(summer-5, experienced-9)
```

Fig. 1. A graphical representation of the Stanford dependencies for the sentence “It was the best summer I have ever experienced.”

- (iv) Modified-by-negative:  $w$  is the head of the dependency, which is modified by a negative word from the prior-polarity lexicon.
- (v) Modifies-intensifier-strengthen:  $w$  modifies a strengthening intensifier from the intensifier lexicon.
- (vi) Modifies-intensifier-weaken:  $w$  modifies a weakening intensifier from the intensifier lexicon.
- (vii) Modified-by-intensifier-strengthen:  $w$  is the head of the dependency, which is modified by a strengthening intensifier from the intensifier lexicon.
- (viii) Modified-by-intensifier-weaken:  $w$  is the head of the dependency, which is modified by a weakening intensifier from the intensifiers lexicon.

**4. Experiments with recognizing the emotion of a sentence**

Our main goal is to evaluate the usefulness of the features described in Section 3 in recognizing the contextual emotion of an emotional word in a sentence. To evaluate those features, we investigate their performance, both together and separately. In the experiments, we use the emotion dataset presented in Section 2. Next, we represent the data with the features presented in Section 3. Those features, however, were defined for each emotional word based on its context; we will therefore proceed differently for sentences with one emotional word and sentences with more than one emotional word.

- (1) In sentences with one emotional word, we assume that the contextual emotion of the emotional word is the same as the emotion assigned to the sentence by the human annotators; therefore all the 625 sentences with one emotional word are represented with the set of features presented in [Section 3](#) and the sentence's emotion will be considered as the contextual emotion of these words.
- (2) For sentences with more than one emotional word, the emotion of the sentence depends on all emotional words and their syntactic and semantic relations. We have 158 sentences where no value can be given to the contextual emotion of their emotional words, and all we know is the overall emotion of the sentence.

We will, therefore, have two different sets of experiments. For the first set of sentences, the data are all annotated, so we will consider supervised learning. In the first set of experiments, we also investigate the performance of our features separately. In these experiments, we group the features defined in [Section 3](#) based on their nature and we assess the usefulness of each group of the features individually. For their evaluation, we use the *t*-test to find out which group contributes to the results significantly. We also look at how the data representation features are related to classes learnt by the machine learning algorithm.

For the second set of sentences, we combine supervised and unsupervised learning. We train a classifier on the first set of data and we use the model to classify the emotional words into their contextual emotion in the second set of data. Finally, we propose an unsupervised method to combine the contextual emotion of all the emotional words in a sentence and calculate the emotion of the sentence. For evaluation, we report precision, recall, *F*-measure and accuracy to compare the results. We also define two baselines for each set of experiments to compare our results with. The experiments are presented in the next three subsections.

#### 4.1. Experiments on sentences with one emotional word

In these experiments, we only work with the sentences which include exactly one emotional word. We first explain the baselines and then the results of our experiments.

**Baselines.** We develop two baseline systems to assess the difficulty of our task. The first baseline labels the sentences the same as the emotion of the most frequent class, which is a typical baseline in machine learning tasks ([Aman and Szpakowicz, 2007](#); [Alm et al., 2005](#)). This baseline will result in 31% accuracy.

The second baseline labels the emotion of the sentence the same as the prior emotion of the only emotional word in the sentence. The accuracy of this experiment is 51%, remarkably higher than the first baseline's accuracy. The second baseline is particularly designed to address the emotion of the sentence only via the prior emotion of the emotional words. It will allow us to assess the difference between the emotion of the sentence based on the prior emotion of the words in the sentence and the emotion which we determine when we consider the context and its effect on the emotion of the sentence.

**Learning experiments.** In this part, we apply two classification algorithms, Support Vector Machines (SVM) and logistic regression (LR), and two different set of features, the set of features from [Section 3](#) and *Bag-of-Words* (unigram). Unigram models have been widely employed in text classification and shown to give good results in sentiment classification tasks.

In general, SVM has long been a method of choice for sentiment recognition in text. SVM has been shown to give good performance in text classification experiments because it scales well to the large numbers of features ([Yang and Li, 1999](#); [Pang et al., 2002](#); [Aman and Szpakowicz, 2007](#)). In the classification, we apply the SMO algorithm ([Platt, 1998](#)) from Weka ([Hall et al., 2009](#)), setting *10-fold cross validation* as a test option. We compare applying SMO to two sets of features: (i) *Bag-of-Words*, which are binary features defining whether a unigram exists in a sentence and (ii) our set of features. In our experiments we use unigrams from the corpus, selected using feature selection methods from Weka.

We also compare those two results with the third experiment: apply SimpleLogistic ([Sumner et al., 2005](#)) from Weka to our set of features, again setting *10-fold cross validation* as a test option. Logistic regression is a discriminative probabilistic classification model which operates over real-valued vector inputs. It is relatively slow to train compared to the other classifiers. It also requires extensive tuning in the form of feature selection and implementation to achieve state-of-the-art classification performance. Logistic regression models with large numbers of features and limited amounts of training data are highly prone to over-fitting ([Alias, 2009](#)). Besides, logistic regression is really slow and it is known to only work on data represented by a small set of features. That is why we do not apply SimpleLogistic to

Table 6

Classification experiments on the dataset with one emotional word in each sentence. Each experiment is marked by the method and the feature set.

		Precision	Recall	<i>F</i>	Accuracy
SVM + <i>Bag-of-Words</i>	Happiness	0.59	0.67	0.63	
	Sadness	0.38	0.45	0.41	
	Anger	0.40	0.31	0.35	
	Surprise	0.41	0.33	0.37	
	Disgust	0.51	0.43	0.47	
	Fear	0.55	0.50	0.52	
	Non-emo	0.49	0.48	0.48	
	Macro-average	0.47	0.45	0.46	50.72%
SVM + our features	Happiness	0.68	0.78	0.73	
	Sadness	0.49	0.58	0.53	
	Anger	0.66	0.48	0.56	
	Surprise	0.61	0.31	0.41	
	Disgust	0.43	0.38	0.40	
	Fear	0.67	0.63	0.65	
	Non-emo	0.51	0.53	0.52	
	Macro-average	0.58	0.53	0.55	58.88%
Logistic regression + our features	Happiness	0.78	0.82	0.80	
	Sadness	0.53	0.64	0.58	
	Anger	0.69	0.62	0.66	
	Surprise	0.89	0.47	0.62	
	Disgust	0.81	0.41	0.55	
	Fear	0.71	0.71	0.71	
	Non-emo	0.53	0.64	0.58	
	Macro-average	0.70	0.61	0.65	66.88%

*Bag-of-Words* features. On the other hand, the number of our features is relatively low, so we find logistic regression to be a good choice of classifier for our representation method. The classification results are shown in Table 6.

We note consistent improvement. The results of both experiments using our set of features significantly outperform (on the basis of a paired *t*-test,  $p = 0.005$ ) both the baselines and SVM applied to *Bag-of-Words* features. We get the best result, however, when we apply logistic regression to our feature set. The number of our features and the nature of the features we introduce make them an appropriate choice of data representation for logistic regression methods.

#### 4.2. Feature set evaluation experiments

In this section, we evaluate the contribution of each group of features to the logistic regression classification results. We perform two series of evaluations. In the first set of evaluations, we use the regression coefficient of the features. The coefficient indicates the nature of the relationship between a particular independent variable and the dependent variable, while a negative value implies an inverse relationship. As a result of applying logistic regression to the sentences with one emotional word, we get the coefficients which are all shown in Table 7 (we only show non-zero values). Although the coefficient values are not a measure of significance per se, we use them to roughly estimate the relation of features with each emotion class. Here, we highlight those relations between the coefficients and each emotion class which we find interesting.

First, we consider the features related to the prior emotion and the prior polarity of the emotional word and their relation with the emotion classes. We will assess the relation of part-of-speech and dependency features later. In the *Sadness* class, there is a positive relation between the emotion class and two features, ‘sad’ and ‘negative’. The ‘sad’ feature indicates whether the prior emotion of the emotional word is sadness; the ‘negative’ feature shows whether the prior polarity of the emotional word is negative based on the lexicon.

Table 7

The coefficients of features with each class. A positive coefficient value indicates a direct relation of the feature and the class, and a negative value implies an inverse relationship.

	[negative]	0.9
	[sad]	2.11
Sadness	[adj]	−1.04
	[RP-0]	−1.14
	[VB-0]	−0.93
	[modifies-negative]	0.71
	[negative]	−0.89
	[positive]	0.66
Happy	[happy]	1.92
	[IN-0]	−1.62
	[DT-1]	0.64
	[DT-2]	0.81
	[neg]	−1.19
	[fear]	2.83
	[surprise]	−0.87
Fear	[CC-2]	1.23
	[FW-1]	1.93
	[RB-2]	−1.34
	[modifies-negative]	0.86
	[length]	0.03
	[negative]	0.74
	[anger]	1.3
Disgust	[disgust]	2.81
	[sad]	−0.63
	[verb]	0.54
	[NNS-1]	1.63
	[TO-1]	−1.03
	[negative]	0.69
	[anger]	2.44
Anger	[fear]	−1
	[EX-2]	2.58
	[WRB-2]	2.08
	[UH-1]	2.36
	[PRP-2]	1
	[negative]	−0.63
	[surprise]	1.29
Surprise	[VB-0]	−0.83
	[IN-1]	0.93
	[TO-1]	1.38
	[CC-2]	1.31
	[modified-by-negative]	−0.74
	[verb]	1.17
	[IN-0]	1.07
Non-Emo	[JJ-1]	0.71
	[NNS-1]	−1.27
	[PRP-1]	−0.78
	[neg]	1.54
	[modified-by-positive]	−0.5

In the *Happiness* class, as expected, there is an inverse relation between the ‘negative’ feature and this class, while it has a direct relation with ‘positive’ and ‘happy’ features.

In the *Fear* class, other than a direct relation with the ‘fear’ feature, it also has an inverse relation with the ‘surprise’ feature. This probably indicates the relation of the two emotion classes, as it can be seen in the sentence “She was shocked to see him with a gun”.

The *Disgust* class has a direct relation with ‘negative’, ‘anger’, and ‘disgust’ features and an inverse relation with ‘sad’. This is, however, in agreement with the fact that Disgust is considered as a challenging and difficult emotion class, which the human annotators often find hard to distinguish from *Anger* (Aman and Szpakowicz, 2007).

The *Anger* class is also in the inverse relation with the ‘fear’ feature, for which we have not found any particular reason.

One interesting relation in the *Surprise* class is its inverse relation with the ‘negative’ feature, which may seem odd if we consider surprise as a negative emotion. A closer look into the dataset, however, shows many sentences such as “my night last night turned out amazing”, which definitely carry a positive surprise.

An investigation of the relation of POS features with each emotion class shows that there is a strong connection, but we do not see any regular pattern worth noting.

Last but not least, we only find very few relations between “dependency” features and emotion classes, which are mainly the polarity and negation modifiers. Particularly the two dependency features, ‘neg’ and ‘modified-by-positive’, can be highlighted for the *Non-Emotional* class. Also, the *Happiness* class has an inverse relation with the ‘neg’ feature which indicated whether the emotional word is modified by a negation word. We can also see that none of the intensity-related features can be observed among the features related with emotion classes; the main reason could be the size of the list (only 112 words) used as an intensifier lexicon, which barely covers any of the words in our dataset. The sparseness of the “dependency” features is problematic. Our observations suggest that very few features in the “dependency” group can occur together in one sentence.

In the second set of experiments meant to evaluate the features, we go more deeply into investigating the significance of the features. For these experiments, we group the features by similarity. The groups are listed below.

- (1) *Prior polarity* indicates whether the prior polarity of the emotional word is positive or negative.
- (2) *Lexical* includes all the features from our lexicons: the prior emotion, the prior polarity, the intensity and the part-of-speech of the emotional word from the emotion lexicon.
- (3) *Part-of-speech* is the same as the part-of-speech features explained in Section 3.
- (4) *Dependency* includes all the features defined as dependency-tree features in Section 3, except the intensity-related features which form a separate group.
- (5) *Negation* includes two negation features. First, is the prior polarity of the emotional word negative? Second, is the emotional word modified by a negative word?
- (6) *Intensity* groups all the features based on the intensifier lexicon. It includes the feature which defines the presence of the emotional word on the intensifier list, and the features which identify the dependency between the emotional word and the words from the intensity lexicon.
- (7) *Length* is the length of the sentence in words.

We perform a series of experiments in which different sets of features are added to the prior-emotion features and new classifiers are trained. Each group of features is applied separately, and the accuracy of the method is calculated. Afterwards, we compare their accuracy with the baseline of 53.44%, which is the classifier’s result only on the prior-emotion feature of the emotional word.

As shown in Table 8, all groups of the features except length contribute positively to the classification result. POS and lexical features, with the accuracy of 60.25% and 57.25% respectively, have the strongest effect and significantly outperform the baseline (on the basis of a paired *t*-test,  $p = 0.001$ ). While dependency and negation features, with the accuracy of 55.20% and 55.90%, improve the results significantly (paired *t*-test,  $p = 0.01$ ), the improvements from the prior polarity, intensity and length of the sentence are not significant.

#### 4.3. Experiments on sentences with more than one emotional word

In this set of experiments, we work with sentences with more than one emotional word, and we combine supervised and unsupervised learning to place each sentence in one of six emotion classes or the non-emotion class. Here, we

Table 8  
*t*-Test results of feature comparison, based on the accuracy of the classifier on each feature set.

Features	Significant ( $p < 0.001$ )	Significant ( $p < 0.01$ )	Not significant	No change
Prior polarity			54.42%	
Lexical	57.25%			
POS	60.25%			
Dependency		55.20%		
Negation		55.99%		
Intensity			53.86%	
Length				53.44%

Table 9  
 Classification experiments on the dataset with more than one emotional word in each sentence. Each experiment is marked by the method and the feature set.

		Precision	Recall	<i>F</i>	Accuracy
SVM + <i>Bag-of-Words</i>	Happiness	0.52	0.60	0.54	
	Sadness	0.35	0.33	0.34	
	Anger	0.30	0.27	0.29	
	Surprise	0.14	0.11	0.12	
	Disgust	0.30	0.17	0.21	
	Fear	0.44	0.29	0.35	
	Non-emo	0.23	0.35	0.28	
	Macro-average	0.32	0.30	0.31	36.71%
Logistic regression + unsupervised + our features	Happiness	0.63	0.71	0.67	
	Sadness	0.67	0.44	0.53	
	Anger	0.50	0.41	0.45	
	Surprise	1.00	0.22	0.36	
	Disgust	0.80	0.22	0.34	
	Fear	0.60	0.64	0.62	
	Non-emo	0.37	0.69	0.48	
	Macro-average	0.65	0.47	0.54	54.43%

have two-step classification. The first step is to learn the classification model from the sentences with one emotion word, explained in Section 2. Next, using that model we get the probability distribution of emotional classes for each emotional word in the sentences with more than one emotional word. The second step is to combine the probability distribution of each emotional word by the unsupervised method; we propose to cluster the sentences into one of the seven classes. While the first step is supervised – using the classifier model learned from the sentences with one emotional word – in the second step the method is unsupervised because there is no training.

**Baselines.** We develop two baseline systems. The first baseline labels all the sentences the same, as the emotion of the most frequent class, giving 32% accuracy. The second baseline labels the emotion of the sentence the same as the most frequently occurring prior-emotion of the emotional words in the sentence. In the case of a tie, we randomly pick one of the emotions. The accuracy of this experiment is 45%. Again, as a second baseline we choose a baseline that is based on the prior emotion of the emotional words so that we can compare it with the results based on contextual emotion of the emotional words in the sentence.

**Learning experiments.** We first train the classifier on the part of the dataset which only includes sentences with one emotional word. For sentences with more than one emotional word, we represent each emotional word and its context by the set of features explained in Section 3. For each emotional word, we get the probability distribution of emotional classes (calculated by the logistic regression function learned from the annotated data in the previous step). We add up the probabilities of each class for all emotional words and select the class with the maximum probability.

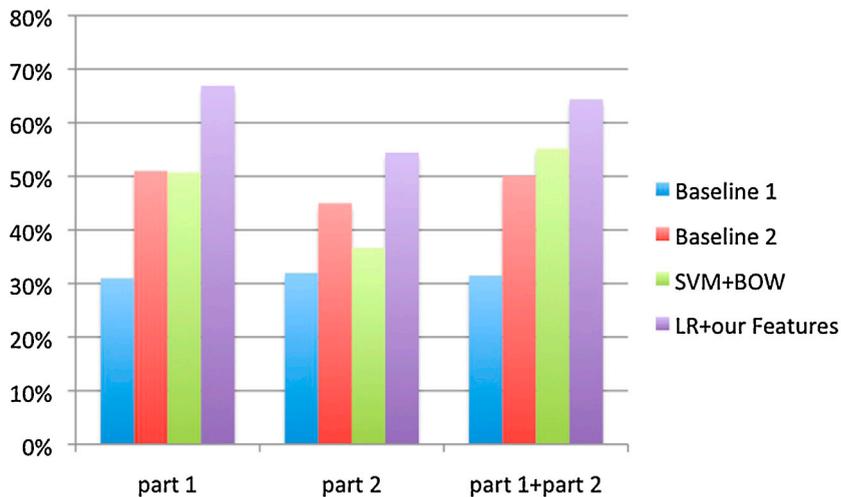


Fig. 2. The comparison of accuracy results of all experiments for sentences with one emotional word (part 1), sentences with more than one emotional words (part 2), and sentences with at least one emotional word (part 1 + part 2).

The result, shown in Table 9, is compared using supervised learning, SVM, with *Bag-of-Words* features, explained in the previous section, with *10-fold cross validation* as a test option.<sup>3</sup>

The results in Table 9 suggest that the result of learning applied to our set of features significantly outperforms (on the basis of a paired *t*-test,  $p = 0.005$ ) both baselines and the result of SVM algorithm applied to *Bag-of-Words* features.

#### 4.4. Discussion

In our experiments, we showed that our method and our features significantly outperform the baselines and the SVM result applied to *Bag-of-Words*. For the final conclusion, we add one more comparison. As we can see from Table 9, the accuracy result of applying SVM to *Bag-of-Words* is really low. Because supervised methods scale well on large datasets, one reason could be the size of the data in this experiment; therefore we try to compare the results of the two experiments on all 783 sentences with at least one emotional word.

For this comparison, we apply SVM with *Bag-of-Words* features to all of 783 sentences and we get an accuracy of 55.17%. Considering our features and methodology, we cannot apply logistic regression with our features to the whole dataset. We calculate its accuracy by counting the percentage of correctly classified instances in both parts of the dataset, used in the two experiments, and we get an accuracy of 64.36%. We also compare the results with the baselines. The first baseline, which is the percentage of most frequent class (Happiness in this case), results in 31.5% accuracy. The second baseline based on the prior emotion of the emotional words results in 50.13% accuracy. It is notable that the result of applying LR to our set of features is still significantly better than the result of applying SVM to *Bag-of-Words* and both baselines; this supports our earlier conclusion. It is hard to compare the results mentioned thus far, so we have combined all the results in Fig. 2, which displays the accuracy obtained by each experiment.

We also looked into our results and assessed the cases where the contextual emotion is different from the prior emotion of the emotional word. Consider the sentence “Joe said it does not happen that often so it does not bother him.” The emotion lexicon classifies the word “bother” as *angry*. This is also the emotion of the sentence if we only consider the prior emotion of words. In our set of features, however, we consider the negation in the sentence, so the sentence is classified as non-emotional rather than angry. Another interesting sentence is the rather simple “You look like her I guess.” Based on the lexicon, the word “like” is in the happy category, while the sentence is non-emotional. In this case, the part-of-speech features play an important role and they catch the fact that “like” is not a verb here. It does not convey a happy emotion, and the sentence is classified as non-emotional.

We also analyzed the errors, and we found some common errors due to:

<sup>3</sup> Since SVM does not return a distribution probability, we cannot apply SVM to our features in this set of experiments.

- (a) limited coverage of the emotion lexicon;
- (b) complex sentences or unstructured sentences which will cause the parser to fail or return incorrect data, resulting in incorrect dependency-tree information.

While we discuss the first issue in the following section, we will address the second issue in our future work.

#### 4.5. Experiments on all emotional sentences

In the previous experiments, we noticed a tight relation between the number of words in each emotion list in the emotion lexicon and the number of sentences derived for each emotion class. Since this suggests that a larger lexicon could have a greater coverage of emotional expressions, we used the *NRC-Emotion* lexicon which covers more emotional words. As explained in Section 2, however, we could not use the *NRC-Emotion* lexicon to train our classifier to learn the contextual emotion of an emotional word. That was due to the lack of sentences with one emotional word (belonging to one emotional class).

On the other hand, one of the weaknesses of our method is the fact that we cannot take advantage of all the instances in the dataset. That is why we could not directly compare our results with the previous results achieved. Again, the main reason is the low coverage of the emotion lexicon we used. The other reason is the limitation of our method: we had only to choose the sentences with one or more emotional words.

To relax these restrictions, we use the *NRC-Emotion* lexicon with a different method of choosing the cue word. In the previous set of experiments, our main focus was on sentences with one emotional word, in order to compare the prior emotion of an emotional word with its contextual emotion. For sentences with one emotional word, we used the only emotional word of the sentence as a cue word. The features were defined for each cue word based on its context. In this set of experiments, most of the sentences are either sentences with more than one emotional word or sentences which have no emotional word. That is why we also found it helpful to add a few sentence-based features. In the following experiments, based on the sentence and the number of emotional words in the sentence, we also have a different strategy of finding the cue word. We have three types of sentences.

- (1) Sentences with more than one emotional word. In these sentences, we will select the cue word based on a simple heuristic assumption that the emotional words which belong to fewer emotion classes are more informative than emotional words which belong to more emotion classes in the lexicon. Also, when there is more than one emotional word which belongs to the same number of emotion classes, we will choose one at random.
- (2) Sentences with one emotional word. In this case we simply consider the only emotional word as a cue word.
- (3) Sentences with no emotional word. In this case we will take the root of the sentence (based on the dependency tree result) as a cue word. We can calculate all the features regarding the root word rather than the emotional word. The root, however, will be emotionally neutral: it belongs to none of the emotion classes in the lexicon. The values of the emotion-word features, explained in Section 3, will be zero.

For this set of experiments, we added the following sentence-based features.

- (a) The number of emotion words in each emotion class according to the *NRC-Emotion* lexicon. This feature is especially useful for sentences with more than one emotion word.
- (b) The number of polar words for each polarity class based on the prior-polarity lexicon. This feature and the next one will give us more information about the sentence. They can be particularly useful for sentences with no emotional word.
- (c) The number of intensifiers in the sentence based on the intensifier lexicon.
- (d) Whether the sentence is in passive voice.
- (e) Whether there is an auxiliary verb in the sentence.
- (f) Whether there is a copula in the sentence.

The last three features were motivated by the work of Wiebe et al. (2005), who state that those features are useful. They found that a sentence tends to be more subjective if it has a copula verb which connects the subject to the complement, and auxiliary verbs which express tense, aspect and modality. They also looked into their training data

Table 10

Classification experiments on the emotional sentences in Aman’s dataset, using the *NRC-Emotion* lexicon and *WordNet-Affect* lexicon respectively.

		Precision	Recall	<i>F</i>	Accuracy
1290 Emo-sentences + NRC lexicon	Happiness	0.66	0.93	0.77	
	Sadness	0.57	0.41	0.48	
	Anger	0.52	0.48	0.50	
	Surprise	0.69	0.30	0.42	
	Disgust	0.52	0.31	0.39	
	Fear	0.56	0.42	0.48	
	Macro-average	0.59	0.47	0.52	61.63%
1290 Emo-sentences + WordNetAffect	Happiness	0.65	0.92	0.76	
	Sadness	0.63	0.50	0.56	
	Anger	0.67	0.49	0.56	
	Surprise	0.64	0.43	0.51	
	Disgust	0.61	0.36	0.45	
	Fear	0.73	0.52	0.61	
	Macro-average	0.65	0.54	0.59	65.04%

and found that the passive voice is often used to query someone about a specific opinion; therefore, they found the active and passive voices to behave very differently. The features are computed from the dependency tree of the sentence. We add these new features to the previous features and we use them all to represent the data. We run the logistic regression classification algorithm, setting *10-fold cross validation* as a test option. Table 10 shows the results.

We want first to validate the application of the extended feature set with the new methodology: we compare its result to both baseline experiments explained in Section 4.1 and also the SVM result applied to *Bag-of-Words*. The first baseline labels the sentences the same as the emotion of the most frequent class, which results in 41.5% accuracy. The second baseline labels the emotion of the sentence the same as the most frequently occurring prior-emotion of the emotional words according to the *NRC-Emotion* lexicon in the sentence. If there is a tie, we randomly pick one of the emotions. The accuracy of this experiment is 51.39%. Finally, SVM’s application to *Bag-of-Words* results in 55.89% accuracy. By comparing these results with our result – the first experiment in Table 10 – we can see a clear improvement in accuracy.

This experiment is also designed to compare our features and our method with the previous work on the whole dataset (Aman and Szpakowicz, 2007). We have two concerns here: first the work in Aman and Szpakowicz (2007) used the whole original dataset (4090 sentences) with all the non-emotional sentences. Due to the high lack of balance we removed 2000 non-emotional sentences, which makes the sizes of the datasets different. So, we cannot directly compare these results.

On the other hand, non-emotional sentences are supposed to have fewer emotional words than other sentences. While we do not quite agree – we saw many non-emotional sentences with emotional words – we must admit that a high percentage of sentences with no emotional words are non-emotional sentences. A closer look at the dataset shows many short sentences such as “Good night!”, “Okay”, and “Yeah” annotated as non-emotional sentence. So, given the different nature of non-emotional sentences, they need a further investigation. Our experiments on the whole dataset explained in Section 2 also emphasize our concern. When using the whole dataset including all non-emotion sentences, we saw a dramatic drop compared to only using the emotional sentences. That brings us to the last experiments, in which we apply our method to all the 1290 emotional sentences on the dataset in Aman and Szpakowicz (2007) using both the *NRC-Emotion* lexicon and *WordNet-Affect* lexicon. The result are shown in Table 10.

We expected an improvement due to a larger lexicon, but the experiment using *WordNet-Affect* improves the result particularly in the negative emotion classes. That could be due to the fact that each word can belong to multiple emotion classes in the *NRC-Emotion* lexicon, which happens quite often among negative emotion classes. We cannot directly compare our results with Aman’s results, because the datasets differ. *F*-measure, precision and recall for each class are reported on the whole dataset, but we only used the emotional sentences of that dataset. To show how hard this task is, and to see where we stand, the best results from Aman and Szpakowicz (2007) are displayed in Table 11.

Table 11

Aman's best results on the whole dataset with 4090 sentences, explained in Section 2.

		Precision	Recall	F
Aman's best results	Happiness	0.81	0.70	0.75
	Sadness	0.60	0.42	0.49
	Anger	0.65	0.44	0.52
	Surprise	0.72	0.41	0.52
	Disgust	0.67	0.49	0.57
	Fear	0.87	0.51	0.64
	Non-emo	0.59	0.62	0.60

## 5. Conclusion and future directions

In this work, we claim that a set of words labeled with their prior emotion is not sufficient for automatic discovery of the emotion of a sentence. The context must also be considered. The focus of this study was a comparison of prior emotion of a word with its contextual emotion, and their effect on the emotion expressed by the sentence. We also studied features important in recognizing contextual emotion. We experimented with a wide variety of linguistically motivated features, and we evaluated the performance of these features using logistic regression. We showed that our method and features significantly outperform the baseline and the SVM result applied to *Bag-of-Words*.

We also grouped the features and evaluate the contribution and significance of each group of features to the results. We found most of the features – lexical, POS, dependency and negation features – improve the results significantly. Even though the features we presented did quite well on the chosen dataset, in the future we would like to show the robustness of these features by applying them to different datasets.

We also used a second lexicon and we found that although the results were slightly better, our features improve the results regardless of the lexicon we choose.

Last but not least, it is a disadvantage that we employ the same set of features regardless of the sentence class. We found that the emotion of the sentence affects its structure and its properties such as tense, length, the presence of negation and so on. It is important to know whether the sentence is neutral or, if it is emotional, whether the emotion is positive or negative. One direction of future work will be to evaluate the effect of our feature sets for each sentence class separately and to try to expand the dependency features based on each emotion class. We are also aware that even more expansion is possible. For example, in the dependency features, we only consider direct dependencies with the emotion word. Now, in sentences such as “there is *no* mother, who does *not* enjoy the success of her child”, the first negation “no” influences the overall emotion of the sentence but is not captured by our features. That is because second-order (indirect) dependencies with the emotion word are ignored. We would like to expand our features to address this omission in the future.

## References

- Alias-i, October 2008. LingPipe 4.1.0. <http://alias-i.com/lingpipe>
- Alm, C.O., Roth, D., Sproat, R., 2005. Emotions from text: machine learning for text-based emotion prediction. In: HLT/EMNLP, pp. 347–354.
- Aman, S., Szpakowicz, S., 2007. Identifying expressions of emotion in text. In: Proc. 10th International Conf. Text, Speech and Dialogue. Springer-Verlag, pp. 196–205.
- Chaumartin, F.-R., 2007. UPAR7: a knowledge-based system for headline sentiment tagging. In: Proceedings of 4th International Workshop on Semantic Evaluations, SemEval'07, pp. 422–425.
- Choi, Y., Cardie, C., Riloff, E., Patwardhan, S., 2005. Identifying sources of opinions with conditional random fields and extraction patterns. In: Proc. Human Language Technology and Empirical Methods in Natural Language Processing, HLT'05, pp. 355–362.
- Ekman, P., 1992. An argument for basic emotions. *Cognition & Emotion* 6 (3), 169–200.
- Esuli, A., Sebastiani, F., 2006. SENTIWORDNET: a publicly available lexical resource for opinion mining. In: Proc. 5th Conf. on Language Resources and Evaluation LREC, pp. 417–422.
- Fahrni, A., Klenner, M., 2008. Old wine or warm beer: target-specific sentiment analysis of adjectives. In: Proc. of the Symposium on Affective Language in Human and Machine, AISB 2008 Convention, pp. 60–63.
- Ghazi, D., Inkpen, D., Szpakowicz, S., 2010. Hierarchical approach to emotion recognition and classification in texts. In: Canadian Conference on Artificial Intelligence, pp. 40–50.

- Hall, M., Frank, E., Holmes, G., Pfahringer, B., Reutemann, P., Witten, I.H., 2009. The WEKA data mining software: an update. *SIGKDD Explorations Newsletter* 11, 10–18.
- Hu, M., Liu, B., 2004. Mining and summarizing customer reviews. In: *Proc. Tenth ACM SIGMOD International Conference on Knowledge Discovery and Data Mining, KDD'04*, pp. 168–177.
- Izard, E.C., 1971. *The Face of Emotion*. Appleton-Century-Crofts, New York.
- Katz, P., Singleton, M., Wicentowski, R., 2007. SWAT-MP: the SemEval-2007 systems for task 5 and task 14. In: *Proc. 4th International Workshop on Semantic Evaluations, SemEval'07*, pp. 308–313.
- Kozareva, Z., Navarro, B., Vázquez, S., Montoyo, A., 2007. UA-ZBSA: a headline emotion classification through web information. In: *Proc. 4th International Workshop on Semantic Evaluations, SemEval'07*, pp. 334–337.
- Liu, B., 2012. *Sentiment Analysis and Opinion Mining*. Morgan & Claypool Publishers.
- Marneffe, M.-C.D., Maccartney, B., Manning, C.D., 2006. Generating typed dependency parses from phrase structure parses. In: *Proc. LREC*, pp. 449–454.
- Mohammad, S.M., Turney, P.D., 2010. Emotions evoked by common words and phrases: using Mechanical Turk to create an emotion lexicon. In: *Proc. NAACL HLT 2010 Workshop on Computational Approaches to Analysis and Generation of Emotion in Text, CAAGET'10*, pp. 26–34.
- Neviarouskaya, A., Prendinger, H., Ishizuka, M., 2010. AM: textual attitude analysis model. In: *Proc. NAACL HLT 2010 Workshop on Computational Approaches to Analysis and Generation of Emotion in Text*, pp. 80–88.
- Neviarouskaya, A., Prendinger, H., Ishizuka, M., 2011. Affect analysis model: novel rule-based approach to affect sensing from text. *Natural Language Engineering* 17 (1), 95–135.
- Ortony, A., Collins, A., Clore, G.L., 1988. *The Cognitive Structure of Emotions*. Cambridge University Press, Cambridge, England.
- Pang, B., Lee, L., Vaithyanathan, S., 2002. Thumbs up? Sentiment classification using machine learning techniques. In: *Proc. 2002 Conference on Empirical Methods in Natural Language Processing, EMNLP'02*, pp. 79–86.
- Parrot, W.G., 2001. *Emotions in Social Psychology*. Psychology Press, Philadelphia, Pennsylvania.
- Picard, R.W., 1997. *Affective Computing*. The MIT Press, Cambridge, Massachusetts.
- Platt, J.C., 1998. Sequential Minimal Optimization: A Fast Algorithm for Training Support Vector Machines. <http://citeseerx.ist.psu.edu/viewdoc/summary?doi=10.1.1.43.4376>
- Plutchik, R., 1962. *The Emotions*. Random House, New York.
- Riloff, E., 2003. Learning extraction patterns for subjective expressions. In: *Proc. 2003 Conf. on Empirical Methods in Natural Language Processing*, pp. 105–112.
- Stoyanov, V., Cardie, C., Wiebe, J., 2005. Multi-perspective question answering using the OpQA corpus. In: *Proc. Conference on Human Language Technology and Empirical Methods in Natural Language Processing, HLT'05*, pp. 923–930.
- Strapparava, C., Mihalcea, R., 2007. SemEval-2007 Task 14: affective text. In: *Proc. Fourth International Workshop on Semantic Evaluations (SemEval-2007)*, Prague, Czech Republic, pp. 70–74.
- Strapparava, C., Mihalcea, R., 2008. Learning to identify emotions in text. In: *Proc. 2008 ACM Symposium on Applied Computing, SAC'08*, pp. 1556–1560.
- Strapparava, C., Valitutti, A., 2004. WordNet-Affect: an affective extension of WordNet. In: *Proc. 4th International Conference on Language Resources and Evaluation*, pp. 1083–1086.
- Sumner, M., Frank, E., Hall, M.A., 2005. Speeding up logistic model tree induction. In: *Proc. 9th European Conference on Principles and Practice of Knowledge Discovery in Databases*, pp. 675–683.
- Toutanova, K., Klein, D., Manning, C.D., Singer, Y., 2003. Feature-rich part-of-speech tagging with a cyclic dependency network. In: *Proc. HLT-NAACL*, pp. 252–259.
- Turney, P.D., 2002. Thumbs up or thumbs down? Semantic orientation applied to unsupervised classification of reviews. In: *Proc. 40th Annual Meeting of the Association for Computational Linguistics, ACL'02*, pp. 417–424.
- Wiebe, J., Wilson, T., Cardie, C., 2005. Annotating expressions of opinions and emotions in language. *Language Resources and Evaluation* 39 (2-3), 165–210.
- Wilson, T., Wiebe, J., Hoffmann, P., 2009. Recognizing contextual polarity: an exploration of features for phrase-level sentiment analysis. *Computational Linguistics* 35 (3), 399–433.
- Yang, Y., Liu, X., 1999. A re-examination of text categorization methods. In: *Proc. 22nd Annual International ACM SIGIR Conference on Research and Development in Information Retrieval, SIGIR'99*, pp. 42–49.
- Yu, H., Hatzivassiloglou, V., 2003. Towards answering opinion questions: separating facts from opinions and identifying the polarity of opinion sentences. In: *Proc. 2003 Conference on Empirical Methods in Natural Language Processing, EMNLP'03*, pp. 129–136.