SU-Sentilab : A Classification System for Sentiment Analysis in Twitter

Gizem Gezici, Rahim Dehkharghani, Berrin Yanikoglu, Dilek Tapucu, Yucel Saygin

Sabanci University

Istanbul, Turkey 34956

{gizemgezici,rdehkharghani,berrin,dilektapucu,ysaygin}@sabanciuniv.edu

Abstract

Sentiment analysis refers to automatically extracting the sentiment present in a given natural language text. We present our participation to the SemEval2013 competition, in the sentiment analysis of Twitter and SMS messages. Our approach for this task is the combination of two sentiment analysis subsystems which are combined together to build the final system. Both subsystems use supervised learning using features based on various polarity lexicons.

1 Introduction

Business owners are interested in the feedback of their customers about the products and services provided by businesses. Social media networks and micro-blogs such as Facebook and Twitter play an important role in this area. Micro-blogs allow users share their ideas with others in terms of small sentences; while Facebook updates may indicate an opinion inside a longer text. Automatic sentiment analysis of text collected from social media makes it possible to quantitatively analyze this feedback.

In this paper we describe our sentiment analysis system identified as SU-Sentilab in the SemEval 2013 competition, Task 2: Sentiment analysis in Twitter. One or the problems in this competition was to label a given tweet or sms message with the correct sentiment orientation, as positive, negative or neutral. In the second task of the same competition, the polarity of a given word or word sequence in the message was asked. Details are described in (Manandhar and Yuret, 2013).

We participated in both of these tasks using a classifier combination consisting of two sub-systems that are based on (Dehkharghani et al., 2012)(Gezici et al., 2012) and adapted to the tweet domain. Both sub-systems use supervised learning in which the system is trained using tweets with known polarities and used to predict the label (polarity) of tweets in the test set. Both systems use features that are based on well-known polarity resources namely SentiWordNet (Baccianella et al., 2010), SenticNet (Cambria et al., 2012) and NRC Emotion Lexicon (Mohammad, 2012). Also a set of positive and negative seed words (Liu et al., 2005) is used in feature extraction.

The remainder of paper is organized as follows: Related works are presented in Section 2; the proposed approach is described in Section 3 and experimental evaluation is presented in Section 4.

2 Related Works

There has been much work on sentiment analysis in the last ten years (Riloff and Wiebe, 2003) (Wilson et al., 2009) (Taboada et al., 2011) (Pang and Lee, 2008). The two fundamental methods for sentiment analysis are lexicon-based and supervised methods. The lexicon-based technique adopts the idea of determining the review sentiment by obtaining word polarities from a lexicon, such as the SentiWordNet (Baccianella et al., 2010), SenticNet (Cambria et al., 2012). This lexicon can be domain-independent or domain-specific. One can use a domain-specific lexicon whenever available, to get a better performance by obtaining the correct word polarities in the given domain (e.g., the word 'small' has a positive mean-

ing in cell phone domain, while it has a negative meaning in hotel domain). On the other hand, establishing a domain-specific lexicon is costly, so many systems use a domain-independent lexicon, such as the SentiWordNet, shortly SWN, (Baccianella et al., 2010) and SenticNet (Cambria et al., 2012). Part of Speech (POS) information is commonly indicated in polarity lexicons, partly to overcome word-sense disambiguity and therefore help achieve a better sentiment classification performance.

Alternatively, supervised methods use machine learning techniques to build models or discriminators for the different classes (e.g. positive reviews), using a large corpus. For example, in (Pang et al., 2002) (Yu and Hatzivassiloglou, 2003), the Naive Bayes algorithm is used to separate positive reviews from negative ones. Note that supervised learning techniques can also use a lexicon in the feature extraction stage. They also generally perform better compared to lexicon-based approaches; however collecting a large training data may be an issue.

In estimating the sentiment of a given natural language text, many issues are considered. For instance one important problem is determining the subjectivity of a given sentence. In an early study, the effects of adjective orientation and gradability on sentence subjectivity was studied (Hatzivassiloglou and Wiebe, 2000). Wiebe et al. (Wiebe et al., 2004) presents a broad survey on subjectivity recognition and the key elements that may have an impact on it.

In estimating the sentiment polarity, the use of higher-order n-grams is also studied. Pang et. al report results where unigrams work better than bigrams for sentiment classification on a movie dataset (Pang et al., 2002). Similarly, occurrence of rare words (Yang et al., 2006) or the position of words in a text are examined for usefulness (Kim and Hovy, 2006)(Pang et al., 2002). In connection with the occurrences of rare words, different variations of delta tf*idf scores of words, indicating the difference in occurrences of words in different classes (positive or negative reviews), have been suggested (Paltoglou and Thelwall, 2010).

In addition to sentiment classification, obtaining the opinion strength is another issue which may be of interest. Wilson et al. (Wilson et al., 2004) for instance, attempts to determine clause-level opinion strength. Since this is a difficult task, one of the re-

cent studies also investigated the relations between word disambiguation and subjectivity, in order to obtain sufficient information for a better sentiment classification (Wiebe and Mihalcea, 2006). A recent survey describing the fundamental approaches can be found in (Liu, 2012).

Two sub-systems combined to form the SU-Sentilab submission are slightly modified from our previous work (Gezici et al., 2012) (Dehkharghani et al., 2012) (Demiroz et al., 2012). For subsystem SU1, we presented some new features in addition to the ones suggested in (Dehkharghani et al., 2012). For subsystem SU2, we combined two systems (Demiroz et al., 2012) (Gezici et al., 2012). The detailed descriptions for our subsystems SU1 and SU2 as well as our combined system can be found in the following sections.

3 System Description

We built two sentiment analysis systems using supervised learning techniques with labelled tweets for training. Then, another classifier was trained for combining the two systems, which is what is submitted to SemEval-2013 Task 2. The subsystems, SU1 and SU2, and also the combination method are explained in the following subsections.

3.1 Subsystem SU1

Subsystem SU1 uses subjectivity based features that are listed in Table 1. These features are divided into two groups:

- F_1 through F_8, using domain independent resources SenticNet (SN) (Cambria et al., 2012), SentiWordNet (SWN) (Baccianella et al., 2010) and the NRC Emotion lexicons (NRC) (Mohammad, 2012),

- F_9 through F_{13} using the seed word list (called SubjWords).

In the remainder of this subsection, we describe the features which are grouped according to the lexical resource used.

SentiWordNet In SentiWordNet (Baccianella et al., 2010), three scores are assigned to each connotation of a word: positivity, negativity and objectivity.

The summation of these three scores equals to one:

$$Pos(w) + Neg(w) + Obj(w) = 1 \qquad (1)$$

where w stands for a given word; and the three scores stand for its positivity, negativity and objectivity scores, respectively. Furthermore, we define the the polarity of a word w as:

$$Pol(w) = Pos(w) - Neg(w) \qquad (2)$$

We also do not do word sense disambiguation (WSD) because it is an ongoing problem that has not been completely solved. The average polarity of all words in a review, r, denoted by $AP(r)$ is computed as in (3).

$$AP(r) = \frac{1}{|r|} \sum_{w_i \in r} Pol(w_i) \qquad (3)$$

where $|r|$ is the number of words in tweet r and $Pol(w_i)$ is the polarity of the word w_i as defined above.

Feature name
F_1: Avg. polarity of all words using SWN
F_2: Avg. polarity of negative words using SWN
F_3: Avg. polarity of positive words using SWN
F_4: Avg. polarity of negative words using SN
F_5: Avg. polarity of positive words using SN
F_6: term frequency of negative words using NRC
F_7: term frequency of positive words using NRC
F_8: term frequency of swear words
F_9: Cumulative frequency of positive SubjWords
F_{10}: Cumulative frequency of negative SubjWords
F_{11}: Proportion of positive to negative SubjWords
F_{12}: Weighted probability of positive SubjWords
F_{13}: Weighted probability of negative SubjWords

Table 1: Features extracted for each tweet in subsystem SU1

The first three features (F_1, F_2, F_3) are based on the average polarity concept (AP). A word w is decided as positive if $Pol(w) > 0$, and decided as negative if $Pol(w) < 0$.

SenticNet   SenticNet (Cambria et al., 2012) is a polarity lexicon that assigns numerical values between -1 and +1 to a *phrase*.

Unlike SentiWordNet (Baccianella et al., 2010), we did not need to do word sense disambiguation for SenticNet. Two features, F_4 and F_5 use the average polarity of negative and positive words extracted from SenticNet. A term is considered as positive if its overall polarity score is greater than 0 and is considered as negative if this score is lower than 0.

NRC Emotion Lexicon   The NRC Emotion Lexicon (Mohammad, 2012) is similar to SenticNet in terms of considering different emotions such as anger and happiness; but it is different from SenticNet because it only assigns a binary value (0 or 1) to words. Features F_6 and F_7 use the number of negative and positive words seen according to this lexicon.

Feature F_8 is an isolated feature from other groups which is the list of English swear words collected from the Internet. As an indication to negative sentiment, we counted the number of appearances of those swear words in tweets and used it as a feature.

Subjective Words (SubjWords)   We also use a set of seed words which is a subset of the seed word list proposed in (Liu et al., 2005), which we called *SubjWords*. The filtering of the original set of subjective words, for a particular domain, is done through a supervised learning process, where words that are not seen in any tweet in the training set are eliminated. Specifically, we add a positive seed word to the positive subset of SubjWords if it has been seen in at least one positive tweet; and similarly a negative seed word is added to negative subset if it has been seen in a negative tweet.

The number of positive and negative words in the initial set before filtering is 2005 and 4783 respectively. Those numbers decrease to 387 positive and 558 negative words after filtering. Note that this filtering helps us to make the seed word sets domain-specific, which in turn helps increase the accuracy of sentiment classification.

The mentioned filtered seed words are used in features F_9 through F_{13} in different ways. For F_9 and F_{10}, we compute the cumulative term frequency of positive and negative seed words for each tweet in the training set, respectively.

$$F_9(r) = \sum_{t_i \in PS} tf(t_i, r) \qquad (4)$$

$$F_{10}(r) = \sum_{t_i \in NS} tf(t_i, r) \qquad (5)$$

The feature F_{11} is the proportion of positive seed words (the number of occurrences) to the negative ones in a review (tweet):

$$F_{11}(r) = \frac{p+1}{n+1} \qquad (6)$$

where p and n are the number of positive and negative seed words, respectively.

Finally features F_{12} and F_{13} are the weighted probabilities of positive and negative words in a review, calculated as follows:

$$F_{12}(r) = p * (1 - P_+(p)) \qquad (7)$$

$$F_{13}(r) = n * (1 - P_-(n)) \qquad (8)$$

where p is the number of positive seed words in review r and $P_+(p)$ is the probability of seeing p positive words in a review. Similarly, $F_{13}(r)$ is the weighted probability of negative words in a review r; n is the number of negative seed words in the review, and $P_-(n)$ is the probability of seeing n negative words in a review. Probabilities $P_+(p)$ and $P_-(n)$ are calculated from training set. Table 2 presents the values of $P_+(p)$ for $n = 1 \ldots 5$. For instance, the probability of seeing at least one positive subjective word in a positive tweet is 0.87, while seeing three positive words is only 0.47.

p	1	2	3	4	5
$P_+(p)$	0.87	0.69	0.47	0.17	0.06

Table 2: The probability of seeing p positive words in a positive tweet.

Classifier   The extracted features are fed into a logistic regression classifier, chosen for its simplicity and successful use in many problems. We have used WEKA 3.6 (Hall et al., 2009) implementation for this classifier, all with default parameters.

3.2 Subsystem SU2

Subsystem SU2 uses word-based and sentence-based features proposed in (Gezici et al., 2012) and summarized in Table 3. For adapting to the tweet domain, we also added some new features regarding smileys.

The features consist of an extensive set of 24 features that can be grouped in five categories: (1) basic features, (2) features based on subjective word occurrence statistics, (3) delta-tf-idf weighting of word polarities, (4) punctuation based features, and (5) sentence-based features. They are as follows:

Basic Features   In this group of features, we exploit word-based features and compute straightforward features which were proposed several times before in the literature (e.g. avg. review polarity and review purity). Moreover, smileys which are crucial symbols in Twitter are also included here.

Seed Words Features   In the second group of features, we have two seed sets as positive and negative seed words. These seed words are the words that are obviously positive or negative irrelevant of the context. As seed words features, we make calculations related to their occurrences in a review to capture several clues for sentiment determination.

Δtf-idf Features   This group consists of features based on the Δtf-idf score of a word-sense pair, indicating the relative occurrence of a word-sense among positive and negative classes (Demiroz et al., 2012).

Punctuation-based Features   This group contains the number of question and exclamation marks in the message, as they may give some information about the sentiment of a review, especially for the Twitter domain.

Sentence-based Features   In this last group of features, we extract features based on sentence type (e.g. subjective, pure, and non-irrealis) (Taboada et al., 2011) and sentence position (e.g. first line and last line) (Zhao et al., 2008). Features include several basic ones such as the average polarity of the first sentence and the average polarity of subjective or pure sentences. We also compute Δtf-idf scores on sentence level.

Finally, we consider the number of sentences which may be significant in SMS messages and the estimated review subjectivity as a feature derived from sentence-level processing. The review is considered subjective if it contains at least one subjec-

tive sentence. In turn, a sentence is subjective if and only if it contains at least one subjective word-sense pair or contains at least one smiley. A word-sense pair is subjective if and only if the sum of its positive and negative polarity taken from SentiWordNet (Baccianella et al., 2010) is bigger than 0.5 (Zhang and Zhang, 2006). These features are described in detail in (Gezici et al., 2012).

Feature name
F_1: Average review polarity
F_2: Review purity
F_3: # of positive smileys
F_4: # of negative smileys
F_5: Freq. of seed words
F_6: Avg. polarity of seed words
F_7: Std. of polarities of seed words
F_8: Δtf-idf weighted avg. polarity of words
F_9: Δtf-idf scores of words
F_{10}: # of Exclamation marks
F_{11}: # of Question marks
F_{12}: Avg. First Line Polarity
F_{13}: Avg. Last Line Polarity
F_{14}: First Line Purity
F_{15}: Last Line Purity
F_{16}: Avg. pol. of subj. sentences
F_{17}: Avg. pol. of pure sentences
F_{18}: Avg. pol. of non-irrealis sentences
F_{19}: Δtf-idf weighted polarity of first line
F_{20}: Δtf-idf scores of words in the first line
F_{21}: Δtf-idf weighted polarity of last line
F_{22}: Δtf-idf scores of words in the last line
F_{23}: Review subjectivity (0 or 1)
F_{24}: Number of sentences in review

Table 3: Features extracted for each tweet in subsystem SU2

Obtaining Polarities from SentiWordNet For all the features in subsystem SU2, we use SentiWord-Net (Baccianella et al., 2010) as a lexicon. Although, we use the same lexicon for our two subsystems, the way we include the lexicon to our subsystems differs. In this subsystem, we obtain the dominant polarity of the word-sense pair from the lexicon and use the sign for the indication of polarity direction. The dominant polarity of a word w, denoted by $Pol(w)$, is calculated as:

$$Pol(w) = \begin{cases} 0 & \text{if } max(p^=, p^+, p^-) = p^= \\ p^+ & \text{else if } p^+ \geq p^- \\ -p^- & \text{otherwise} \end{cases}$$

where p^+, $p^=$ and p^- are the positive, objective and negative polarities of a word w, respectively.

After obtaining the dominant polarities of words from SentiWordNet (Baccianella et al., 2010), we update these polarities using our domain adaptation technique (Demiroz et al., 2012). The $\Delta tf - idf$ scores of words are computed and if there is a disagreement between the $\Delta tf - idf$ and the dominant polarity of a word indicated by the lexicon, then the polarity of the word is updated. This adaptation is described in detail in one of our previous works (Demiroz et al., 2012).

Classifier The extracted features are fed into a Naive Bayes classifier, also chosen for its simplicity and successful use in many problems. We have used WEKA 3.6 (Hall et al., 2009) implementation for this classifier, where the Kernel estimator parameter was set to true.

3.3 Combination of Subsystems

As we had two independently developed systems that were only slightly adapted for this competition, we wanted to apply a sophisticated classifier combination technique. Rather than averaging the outputs of the two classifiers, we used the development set to train a new classifier, to learn how to best combine the two systems. Note that in this way the combiner takes into account the different score scales and accuracies of the two sub-systems automatically.

The new classifier takes as features the probabilities assigned by the systems to the three possible classes (positive, objective, negative) and another feature which is an estimate of subjectivity of the tweet or SMS messages. We trained the system using these 7 features obtained from the development data for which we had the groundtruth, with the goal of predicting the actual class label based on the estimates of the two subsystems.

4 Evaluation

4.1 Competition Tasks

There were two tasks in this competition: 1) Task A where the aim was to determine the sentiment of a phrase within the message and 2) Task B where the aim was to obtain the overall sentiment of a message. In each task, the classification involves the assignment of one of the three sentiment classes, positive, negative and objective/neutral. There were two different datasets for each task, namely tweet and SMS datasets (Manandhar and Yuret, 2013). Due to the different nature of tweets and SMS and the two tasks (A and B), we in fact considered this as four different tasks.

4.2 Submitted Systems

Due to time constraints, we mainly worked on TaskB where we had some prior experience, and only submitted participated in TaskA for completeness.

As we did not use any outside labelled data (tweets or SMS), we trained our systems on the available training data which consisted only of tweets and submitted them on both tweets and SMS sets. In fact, we separated part of the training data as validation set and comparison of the two subsystems.

Since only one system is allowed for each task, we selected the submitted system from our 3 systems (SU1, SU2, combined) based on their performance on the validation set. The performances of these systems are summarized in Table 4.

Finally, we re-trained the selected system with the full training data, to use all available data.

For the implementation, we used C# for subsystem SU1 and Java & Stanford NLP Parser (De Marneffe and Manning, 2008) for subsystem SU2 and WEKA (Hall et al., 2009) for the classification part for both of the systems.

4.3 Results

In order to evaluate and compare the performances of our two systems, we separated a portion of the training data as validation set, and kept it separate. Then we trained each system on the training set and tested it on the validation set. These test results are given in Table 4.

We obtained 75.60% accuracy on the validation set with subsystem SU1 on TaskA_twitter using logistic regression. For the same dataset, we obtained 70.74% accuracy on the validation set with subsystem SU2 using a Naive Bayes classifier.

For TaskB_Twitter dataset on the other hand, we benefited from our combined system in order to get better results. With this combined system using logistic regression as a classifier, we achieved 64% accuracy on the validation set. The accuracies obtained by the individual subsystems on this task was 63.10% by SU1 and 62.92% by SU2.

Dataset	System	Accuracy
TaskA_Twitter	SU1	75.60%
	SU2	70.74%
TaskB_Twitter	SU1	63.10%
	SU2	62.92%
	Combined	64.00%

Table 4: Performance of Our Systems on Validation Data

4.4 Discussion & Future Work

The accuracy of our submitted systems for different tasks are not very high due to many factors. First of all, both domains (tweets and SMSs) were new to us as we had only worked on review polarity estimation on hotel and movie domains before.

For tweets, the problem is quite difficult due to especially short message length; misspelled words; and lack of domain knowledge (e.g. 'Good Girl, Bad Girl' does not convey a sentiment, rather it is a stage play's name). As for the SMS data, there were no training data for SMSs, so we could not tune or retrain our existing systems, either. Finally, for Task A, we had some difficulty with the phrase index, due to some ambiguity in the documentation. Nonetheless, we thank the organizers for a chance to evaluate ourselves among others.

This was our first experience with this competition and with the Twitter and SMS domains. Given the nature of tweets, we used simple features extracted from term polarities obtained from domain-independent lexicons. In the future, we intend to use more sophisticated algorithms, both in the natural language processing stage, as well as the machine learning algorithms.

References

Stefano Baccianella, Andrea Esuli, and Fabrizio Sebastiani. 2010. Sentiwordnet 3.0: An enhanced lexical resource for sentiment analysis and opinion mining. In *Proc. of LREC*.

Erik Cambria, Catherine Havasi, and Amir Hussain. 2012. Senticnet 2: A semantic and affective resource for opinion mining and sentiment analysis. In G. Michael Youngblood and Philip M. McCarthy, editors, *FLAIRS Conf.* AAAI Press.

Marie-Catherine De Marneffe and Christopher D Manning. 2008. Stanford typed dependencies manual. *URL http://nlp. stanford. edu/software/dependencies_manual. pdf*.

Rahim Dehkharghani, Berrin Yanikoglu, Dilek Tapucu, and Yucel Saygin. 2012. Adaptation and use of subjectivity lexicons for domain dependent sentiment classification. In *Data Mining Workshops (ICDMW), 2012 IEEE 12th International Conf.*, pages 669–673.

Gulsen Demiroz, Berrin Yanikoglu, Dilek Tapucu, and Yucel Saygin. 2012. Learning domain-specific polarity lexicons. In *Data Mining Workshops (ICDMW), 2012 IEEE 12th International Conf. on*, pages 674–679.

Gizem Gezici, Berrin Yanikoglu, Dilek Tapucu, and Yücel Saygın. 2012. New features for sentiment analysis: Do sentences matter? In *SDAD 2012 The 1st International Workshop on Sentiment Discovery from Affective Data*, page 5.

Mark Hall, Eibe Frank, Geoffrey Holmes, Bernhard Pfahringer, Peter Reutemann, and Ian H Witten. 2009. The weka data mining software: an update. *ACM SIGKDD Explorations Newsletter*, 11(1):10–18.

Vasileios Hatzivassiloglou and Janyce M Wiebe. 2000. Effects of adjective orientation and gradability on sentence subjectivity. In *Proc. of the 18th Conf. on Comp. Ling.-Volume 1*, pages 299–305.

Soo-Min Kim and Eduard Hovy. 2006. Automatic identification of pro and con reasons in online reviews. In *Proc. of the COLING/ACL Main Conf. Poster Sessions*, pages 483–490.

Bing Liu, Minqing Hu, and Junsheng Cheng. 2005. Opinion observer: analyzing and comparing opinions on the web. In *WWW '05: Proc. of the 14th International Conf. on World Wide Web*.

Bing Liu. 2012. Sentiment analysis and opinion mining. *Synthesis Lectures on Human Language Technologies*, 5(1):1–167.

Suresh Manandhar and Deniz Yuret. 2013. Semeval tweet competition. In *Proc. of the 7th International Workshop on Semantic Evaluation (SemEval 2013) in conjunction with the Second Joint Conference on Lexical and Comp.Semantics (*SEM 2013)*.

Saif Mohammad. 2012. #emotional tweets. In **SEM 2012: The First Joint Conf. on Lexical and Comp. Semantics*, pages 246–255. Association for Comp. Ling.

Georgios Paltoglou and Mike Thelwall. 2010. A study of information retrieval weighting schemes for sentiment analysis. In *Proc. of the 48th Annual Meeting of the Association for Comp. Ling.*, pages 1386–1395.

Bo Pang and Lillian Lee. 2008. Opinion mining and sentiment analysis. *Foundations and Trends in Information Retrieval*, 2(1-2):1–135, January.

Bo Pang, Lillian Lee, and Shivakumar Vaithyanathan. 2002. Thumbs up? sentiment classification using machine learning techniques. In *Proc. of EMNLP*, pages 79–86.

Ellen Riloff and Janyce Wiebe. 2003. Learning extraction patterns for subjective expressions. In *Proc. of the 2003 Conf. on Empirical methods in natural language processing*, pages 105–112, Morristown, NJ, USA. Association for Comp. Ling.

Maite Taboada, Julian Brooke, Milan Tofiloski, Kimberly Voll, and Manfred Stede. 2011. Lexicon-based methods for sentiment analysis. *Comput. Linguist.*, 37(2):267–307.

Janyce Wiebe and Rada Mihalcea. 2006. Word sense and subjectivity. In *ACL*.

Janyce Wiebe, Theresa Wilson, Rebecca F. Bruce, Matthew Bell, and Melanie Martin. 2004. Learning subjective language. *Comp. Ling.*, 30(3):277–308.

Theresa Wilson, Janyce Wiebe, and Rebecca Hwa. 2004. Just how mad are you? finding strong and weak opinion clauses. In *Proc. of the 19th national Conf. on Artifical intelligence*, AAAI'04, pages 761–767.

Theresa Wilson, Janyce Wiebe, and Paul Hoffmann. 2009. Recognizing contextual polarity: An exploration of features for phrase-level sentiment analysis. *Comp. Ling.*, pages 399–433.

Kiduk Yang, Ning Yu, Alejandro Valerio, and Hui Zhang. 2006. Widit in trec-2006 blog track. In *Proc. of TREC*, pages 27–31.

Hong Yu and Vasileios Hatzivassiloglou. 2003. Towards answering opinion questions: Separating facts from opinions and identifying the polarity of opinion sentences. In *Proc. of the 2003 Conf. on Empirical methods in natural language processing*, pages 129–136. Association for Comp. Ling.

Ethan Zhang and Yi Zhang. 2006. Ucsc on trec 2006 blog opinion mining. In *Text Retrieval Conference*.

Jun Zhao, Kang Liu, and Gen Wang. 2008. Adding redundant features for crfs-based sentence sentiment classification. In *Proc. of the conference on empirical methods in natural language processing*, pages 117–126. Association for Comp. Ling.