Effect of Using Regression on Class Confidence Scores in Sentiment Analysis of Twitter Data

Itir Onal[*], Ali Mert Ertugrul[†], Ruken Cakici[*]
[*]Department of Computer Engineering, Middle East Technical University, Ankara, Turkey
itir,ruken@ceng.metu.edu.tr
[†]Department of Information Systems, Middle East Technical University, Ankara, Turkey
alimert@metu.edu.tr

Abstract

In this study, we aim to test our hypothesis that confidence scores of sentiment values of tweets aid in classification of sentiment. We used several feature sets consisting of lexical features, emoticons, features based on sentiment scores and combination of lexical and sentiment features. Since our dataset includes confidence scores of real numbers in [0-1] range, we employ regression analysis on each class of sentiments. We determine the class label of a tweet by looking at the maximum of the confidence scores assigned to it by these regressors. We test the results against classification results obtained by converting the confidence scores into discrete labels. Thus, the strength of sentiment is ignored. Our expectation was that taking the strength of sentiment into consideration would improve the classification results. Contrary to our expectations, our results indicate that using classification on discrete class labels and ignoring sentiment strength perform similar to using regression on continuous confidence scores.

1 Introduction

In the past few years, there has been a growing interest in using microblogging sites such as Twitter. Generally, people tend to share their opinions, ideas about entities, topics and issues via these microblogs. Therefore, companies show interest in these for the sentiment analysis to be used as means of customer satisfaction evaluation about their products.

Although some tweets express direct sentiment, the polarity and intent of some tweets cannot be understood even by humans because of lack of context. Moreover, a tweet may be perceived as positive or negative by some people whereas others may think that the tweet is not polar. Therefore, sometimes it is not easy to assign a sentiment class to a tweet. Instead of assigning a single sentiment to a tweet, confidence scores reflecting the likelihoods of sentiments of the tweet may be provided. Our dataset consists of tweets and their corresponding confidence scores of five sentiments namely *positive, negative, neutral, irrelevant* and *unknown*. An analysis on the dataset reflects that, some tweets get similar confidence scores for many classes. In other words, different people assign different class labels to the same tweet. On the other hand, confidence scores of some tweets for a class are close to or equal to 1, meaning that the sentiment of the tweets are clear. If we have discrete class labels for all tweets, tweets assigned to classes with a low confidence score will have equal effect as the ones whose confidence scores are high during the training phase of sentiment analysis.

In this study, we investigate whether the strength of sentiment plays a role in classification or not. We build regression models to estimate the confidence scores of tweets for each class separately. Then, we assign the sentiment, whose confidence score is maximum among others to the tweet. On the other hand, we also converted the confidence scores to discrete class labels and performed classification directly. The experiments and results are explained in Section 5.

2 Related Work

Sentiment analysis on Twitter has some challenges compared to the classical sentiment analysis methods on formal documents since the tweets may have irregular structure, short length and non-English words. Moreover, they may include elements specific to microblogs such as hashtags, emoticons, etc. Go et al. (2009) used emoticons as features and Barbosa et al. (2010) used

Proceedings of the 5th Workshop on Computational Approaches to Subjectivity, Sentiment and Social Media Analysis, pages 136–141,
Baltimore, Maryland, USA. June 27, 2014.

retweets, hashtags, emoticons, links, etc. as features to classify the sentiments as positive or negative. Furthermore, Kouloumpis et al. (2011) showed that the features including presence of intensifiers, positive/negative/neutral emoticons and abbreviations are more successful than part-of-speech tags for sentiment analysis on Twitter. Saif et al. (2012) extracted sentiment topics from tweets and then used them to augment the feature space. Agarwal et al. (2011) used tree kernel to determine features and they used SVM, Naïve Bayes, Maximum entropy for classification. In our experiments we used k-Nearest Neighbor (k-NN) and SVM as classifiers.

Due to the rarity of class confidence scores of datasets in the literature, a few studies employ regression. Jain et al. (2012) use Support Vector Regression (SVR) for sentiment analysis in movie reviews but the labels they use are discrete. So, they use SVR directly for classification purpose, not regression. However, we employed SVR on confidence scores with the aim of regression. Moreover, Lu et al. (2011) use SVR in multi-aspect sentiment analysis to detect the ratings of each aspect. Since our approach does not include aspects, our results are not comparable with that of (Lu et al., 2011). The study of Liu (2012) consists of studies employing regression in sentiment analysis. Yet, in most of these studies the regressors are trained using discrete rating scores between 1 and 5. Furthermore, Pang et al. (2008) also mentions regression to classify sentiments using discrete rating scores. Unlike these approaches, we employ regression on real-valued confidence scores between 0 and 1.

3 Data Description and Pre-processing

The data set we use (Kaggle, 2013) consists of 77946 tweets which are obtained with the aim of sentiment classification. Each tweet is rated by multiple raters and as a result, each tweet has confidence scores of five classes namely *positive, negative, neutral, irrelevant* and *unknown*. Among 77946 tweets, only 800 of them has the maximum confidence score of *unknown* class. Therefore, in order to have a balanced dataset in our experiments, we selected 800 tweets from each class. As a result, the dataset used in our experiments is balanced and includes a total of 4000 tweets.

The data set includes tweets both relevant and irrelevant to weather. Tweets are expected to get high confidence score of irrelevant class if the tweet is not related to weather. Moreover, as their name implies, positive and negative confidence values represent the polarity level of each tweet towards weather. If a tweet is not polar, it is expected to be given a high neutral confidence score. Unknown class is expected to have a high score when the tweet is related with weather, but the polarity of tweet cannot be decided.

The tweets in the data set are labeled by multiple raters. Then, the confidence scores for labels are obtained by aggregating labels given to tweets by raters and the individual reliability of each rater. For a tweet, confidence scores of all categories sum to 1 and confidence score values are in range [0,1].

Before feature extraction, we pre-process the data in a few steps. Firstly, we remove *links* and *mentions* that are features specific to tweets. Then, we remove *emoticons* from the text while recording their number for each tweet in order to use them later.

4 Features

Our features can be divided into four main categories which are lexical features, emoticons, features based on sentiment scores and a combination of the lexical and sentiment features.

4.1 Lexical Features

We extracted two different lexical features which are word n-grams, part-of-speech (POS) n-grams. Using all tweets in our training data, we extracted only unigrams of words to be used as baseline. Moreover, after extracting POS tags of sentences in each tweet using the POS tagger given in (Toutanova et al., 2003), we computed unigrams and bigrams of POS tags. We considered the presence of word unigrams, POS unigrams and bigrams. Therefore, those features can get binary values.

4.2 Emoticons

In the preprocessing step, we remove the emoticons from the text. However, since emoticons carry sentiment information, we also record whether the tweet includes positive, negative or neutral emoticons (see Table 1) during the removal of emoticons. Therefore, we extract 3 binary features based on emoticon presence in the tweet.

Table 1: Emoticons and their sentiments

Sentiment	Emoticon
Positive	:) , :-), =), =D, :D
Negative	:(, :-(, =(, :/
Neutral	:|

4.3 Features Based on Sentiment Scores

We extract features based on sentiment scores using two different approaches. In the first one, we use SentiWordNet 3.0 (Baccianella et al., 2010) to obtain the sentiment scores of each word. We used the word and a tag representing the POS tag of the word to output the sentiment score of the word. Since the same word with different senses have different scores, we obtained a single sentiment score by computing the weighted average of SentiWordNet scores for each sense. Furthermore, POS tagging is performed as explained in 4.1. However, since POS tags of Penn TreeBank and SentiWordNet are different, we convert one to other as shown in Table 2. Therefore, the sentiment score for a word is obtained after the Penn TreeBank tags are converted to SentiWordNet tags. Using all the words in a tweet and their corresponding SentiWordNet scores, we compute the following features:

- # of words having positive sentiment
- # of words having negative sentiment
- total sentiment score

As a result, using SentiWordNet, we extract 3 more features. We observe that the acronym *lol* representing *laughing out loud* is used extensively in tweets. In order to keep its meaning, when a *lol* is encountered, its sentiment score is assigned to 1. Moreover, sentiment scores of words having other POS tags than the ones in Table 2 are assigned to 0. When *not* is encountered, we multiply the sentiment score of its successor word by -1 and convert the sentiment score of *not* to 0.

Table 2: Conversion of POS tags to SentiWordNet tags

SentiWordNet Tag	Penn TreeBank tag
a (adjective)	JJ, JJR, JJS
n (noun)	NN, NNS, NNP, NNPS
v (verb)	VB,VBD, VBG, VBN, VBP, VPZ
r (adverb)	RB, RBR, RBS

The second approach is using LabMT word list (Dodds et al., 2011) which includes scores for sentiment analysis. It includes a list of words with their happiness rank, happiness average and happiness standard deviation. In our study, we computed those values for all the words in a tweet and extracted the 6 features namely the minima and the maxima of happiness rank, happiness average and happiness standard deviation.

Note that, if a word is not encountered in either SentiWordNet or labMT dictionary, then the sentiment score of that word is assigned to 0.

4.4 Combination of Lexical and Sentiment Features

We extract features using POS tags and sentiment scores. After the conversion of POS tags in Table 2, we have four main tags namely, ***a*** *(adjective),* ***n*** *(noun),* ***v*** *(verb),* ***r*** *(adverb).* For each tweet we compute the number of adjectives, nouns, verbs and adverbs having positive, negative and neutral sentiments. Therefore, we extract 12 features using combination of lexical and sentiment features. Table 3 shows all the features used.

5 Experiments

In our experiments we extract the features using training data set. Then, we formed training and test feature matrices using these features. By using these matrices, we both conduct classification and regression.

We train separate regressors for each class using the training feature matrix and confidence scores of the corresponding class. We use Support Vector Regression (SVR) library of (Chang et al., 2011) in our computations. Recall that, the confidence scores are between 0 and 1 and they carry information about how likely it is that a tweet belongs to a specified class. For instance it is very likely that a positive with a 0.9 confidence score is actually a positive, whereas a positive with a 0.2 confidence score is much less likely to be positive. In order to assign a sentiment label to a test tweet, we separately test that tweet with the regressors trained for each class. Then, each regressor assigns a score between 0 and 1 to that test tweet. Finally, we assign the class label with maximum score to the test tweet.

During classification, we convert confidence scores to discrete class labels by assigning them the class which the majority of the raters agreed upon. Using training feature matrix and their corresponding discrete labels, we train a Support Vec-

Table 3: Features used in our experiments

Lexical	word unigram	f_1
	POS unigram + bigram	f_2
Emoticons	# of pos, neg, neu emoticons	f_3
Sentiment Scores	SentiWordNet (# of pos, neg words, total sentiment score)	f_4
	labMT (min, max of happiness rank, avg and std)	f_5
Sentiment + Lexical	# of pos a, pos n, pos v, pos r # of neg a, neg n, neg v, neg r # of neu a, neu n, neu v, neu r	f_6

tor Machine (SVM) using the method of (Chang et al., 2011) and a k-Nearest Neighbor (k-NN) classifier. SVM and k-NN directly assigns class labels to test tweets.

We employed classification and regression on three types of data having classes:

- positive - negative - neutral - irrelevant - unknown
- positive - negative - neutral
- positive - negative

5.1 Positive vs. Negative vs. Neutral vs. Irrelevant vs. Unknown

In 5-class classification, our dataset consists of 4000 tweets (800 for each class). We used 3000 of them as training data (600 for each class) and 1000 of them as test data (200 for each class). Since our dataset is balanced, chance accuracy is 20% if we assing all the tweets to one class. Using various features to train k-NN, SVM and SVR, we obtained the results in Table 4.

Results in Table 4 show that, classification with SVM performs the best when emoticon features (f_3) and SentiWordNet features (f_4) are combined with unigram baseline. Moreover, using emoticon features (f_3), and sentiment score features (both SentiWordNet (f_4) and labMT (f_5)) together with the word unigram baseline perform the best among others when SVR is used. Notice that using regression performs slightly worse than using SVM for most of the feature combinations. However, the p-value of SVM vs. SVR is 0.06, meaning that the performance improvement of SVM is insignificant. On the other hand, using SVR always performs much better than k-NN with a p-value of 2×10^{-10}.

5.2 Positive vs. Negative vs. Neutral

In 3-class classification, our dataset consists of 2400 tweets (800 for each class). We use 1800 of them as training data (600 for each class) and 600 of them as test data (200 for each class). Since our dataset is balanced, chance accuracy is 33%. Using various features to train k-NN, SVM and SVR, we obtain the results in Table 5.

Table 4: k-NN, SVM and SVR Performances for 5-class classification

Features	**k-NN**	**SVM**	**SVR**
Unigram (f_1)	0,3140	0,4430	0,4290
$+f_2$	0,3130	0,4330	0,4300
$+f_3$	0,3350	0,4410	0,4350
$+f_5$	0,3280	0,4460	0,4340
$+f_6$	0,3490	0,4500	0,4260
$+f_3, f_4$	0,3450	**0,4570**	0,4370
$+f_3, f_5$	0,3300	0,4430	0,4340
$+f_4, f_5$	**0,3550**	0,4490	0,4350
$+f_4, f_6$	0,3490	0,4550	0,4260
$+f_3, f_4, f_5$	0,3530	0,4490	**0,4430**
$+f_2, f_3, f_4, f_5$	0,3500	0,4350	0,4420
$+f_2, f_3, f_4, f_5, f_6$	0,3430	0,4250	0,4350

Table 5: k-NN, SVM and SVR Performances for 3-class classification

Features	**k-NN**	**SVM**	**SVR**
Unigram (f_1)	0,5183	0,6650	0,6467
$+f_2$	0,5017	0,6267	0,6450
$+f_3$	0,5333	**0,6767**	0,6567
$+f_5$	0,5467	0,6617	0,6533
$+f_6$	0,5450	**0,6767**	**0,6700**
$+f_3, f_4$	0,5550	0,6717	0,6583
$+f_3, f_5$	0,5517	0,6700	0,6667
$+f_4, f_5$	0,5533	0,6733	0,6567
$+f_4, f_6$	0,5233	0,6750	**0,6700**
$+f_3, f_4, f_5$	**0,5700**	0,6700	0,6550
$+f_2, f_3, f_4, f_5$	0,5367	0,6583	0,6567
$+f_2, f_3, f_4, f_5, f_6$	0,5450	0,6500	0,6550

Table 5 reflects that, using the combination of sentiment and lexical features (f_6) play an important role in positive - negative - neutral classification using SVR. On the other hand, using emoticon features (f_3) with unigram baseline or labMT features (f_5) with unigram baseline performs the best when SVM is used. It can be seen that SVM performs slightly better than SVR most of the time yet the performance improvemen is again insignificant with a p-value of 0.58. Furthermore, they always perform much better than k-NN with a p-value of 2×10^{-8}.

5.3 Positive vs. Negative

In 2-class classification, since we have 800 positive and 800 negative tweets among 4000 tweets, we used 1600 tweets. We used 1200 of them as training data (600 for each class) and 400 of them as test data (200 for each class). Since our dataset is balanced, chance accuracy is 50%. Using the same set of features to train k-NN, SVM and SVR, we obtained the results in Table 6.

Table 6: k-NN, SVM and SVR Performances for 2-class classification

Features	**k-NN**	**SVM**	**SVR**
Unigram (f_1)	0,6275	0,7700	0,7775
$+f_2$	0,6575	0,7575	0,7375
$+f_3$	**0,7225**	0,7850	0,7775
$+f_5$	0,6900	0,7575	0,7700
$+f_6$	0,6950	**0,7975**	**0,7975**
$+f_3, f_4$	0,6900	0,7825	0,7850
$+f_3, f_5$	0,7125	0,7800	0,7700
$+f_4, f_5$	0,6950	0,7800	0,7800
$+f_4, f_6$	0,6725	0,7950	**0,7975**
$+f_3, f_4, f_5$	0,7000	0,7725	0,7800
$+f_2, f_3, f_4, f_5$	0,6675	0,7700	0,7800
$+f_2, f_3, f_4, f_5, f_6$	0,6675	0,7825	0,7750

In positive - negative classification, using combination of sentiment and lexical features (f_6) with unigram baseline results in the highest performance among all when either SVM or SVR is used. Similar to previous classification results, performance improvement of using SVM on discrete labels instead of using SVR is insignificant with a p-value of 0.46 whereas SVR provides a significant performance improvement over k-NN with a p-value of 5×10^{-4}.

6 Conclusion

In this study we conducted sentiment analysis on tweets about weather. We performed two types of experiments, one using confidence scores directly by regression and the other one by discreticising this information and using discrete classifiers. We expected that employing regression on confidence scores would better discriminate the sentiment classes of tweets than the classification on discrete labels since they consider the sentiment strength.

First, we extracted various types of features including lexical features, emoticons, sentiment scores and combination of lexical and sentiment features. Then, we created the feature vectors for these tweets. We trained a regressor for each class separately using continuous valued confidence scores. Then, a test tweet is assigned to the label, whose estimated confidence score is the highest among others. In our second experiment, we assigned class labels having the maximum confidence score to the tweets in the training set directly. Using the training data and discrete valued class labels, we trained a classifier. Then, a test tweet is assigned to a class label by the classifier.

Our results indicate that using classification on discrete valued class labels performs slightly better than using regression, which considers confidence scores during training. However, the performance improvement is shown to be insignificant. We would expect a significant performance improvement using SVR compared to SVM as in the case of k-NN vs. SVR. However, we explored that the effect of strength of sentiment is insignificant.

As future work, we will employ our methods on datasets including continuous scores rather than discrete class labels such as movie reviews including ratings. Moreover, we may enhance our approach on multi-aspect sentiment analysis problems where each aspect is given ratings.

References

Alec Go, Richa Bhayani and Lei Huang. 2009. Twitter sentiment classification using distant supervision. *Stanford Digital Library Technologies Project, NJ*.

Luciano Barbosa and Junlan Feng 2010. Robust Sentiment Detection on Twitter from Biased and Noisy Data. *Proceedings of COLING, Beijing China*, 36–44.

Efthymios Kouloumpis, Theresa Wilson and Johanna Moore 2011. Twitter Sentiment Analysis: The Good the Bad and the OMG! *Proceedings of the ICWSM, Barcelona, Spain*.

Hassan Saif, Yulan He, and Harith Alani 2012. Alleviating data sparsity for twitter. *2nd Workshop on Making Sense of Microposts, Lyon, France*.

Apoorv Agarwal, Boyi Xie, Ilia Vovsha, Owen Rambow and Rebecca Passonneau 2011. Sentiment analysis of twitter data. *Proceedings of the Workshop on Languages in Social Media, Portland, Oregon, USA*, 30–38

Siddharth Jain and Sushobhan Nayak 2012. Sentiment Analysis of Movie Reviews: A Study of Features and Classifiers. *CS221 Course Project: Artificial Intelligence , Stanford (Fall 2012) [Report]*.

Bin Lu, Myle Ott, Claire Cardie, and Benjamin Tsou 2011. Multi-aspect sentiment analysis with topic models. *The ICDM2011 Workshop on Sentiment Elicitation from Natural Text for Information Retrieval and Extraction, Vancouver, Canada*.

Kristina Toutanova, Dan Klein, Christopher Manning, and Yoram Singer. 2003. Feature-Rich Part-of-Speech Tagging with a Cyclic Dependency Network. *Proceedings of HLT-NAACL 2003, Edmonton, Canada*, 252–259.

Stefano Baccianella, Andrea Esuli, and Fabrizio Sebastiani 2010. SentiWordNet 3.0: An Enhanced Lexical Resource for Sentiment Analysis and Opinion Mining *Proceedings of the Seventh International Conference on Language Resources and Evaluation (LREC'10), Valletta, Malta*

Peter S. Dodds, Kameron D.Harris, Isabel M. Kloumann, Catherine A. Bliss and Christopher M. Danforth 2011. Temporal Patterns of Happiness and Information in a Global Social Network: Hedonometrics and Twitter *PLoS ONE 6(12): e26752*

Chih-Chung Chang and Chih-Jen Lin 2011. LIBSVM: A Library for Support Vector Machines *ACM Transactions on Intelligent Systems and Technology*, 2:27:1–27:27

Kaggle "Partly Sunny with a Chance of Hashtags" competition dataset 2013 http://www.kaggle.com/c/crowdflower-weather-twitter

Quinn McNemar 1947 Note on the sampling error of the difference between correlated proportions or percentages *Psychometrika* 12(2):153-157

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

Bing Liu 2012 Sentiment Analysis and Opinion Mining *Morgan & Claypool Publishers*