NileTMRG at SemEval-2016 Task 7: Deriving Prior Polarities for Arabic Sentiment Terms

Samhaa R. El-Beltagy
Center for Informatics Sciences
Nile University, Juhayna Square, Sheikh Zayed City
Giza, Egypt
samhaa@computer.org

Abstract

This paper presents a model that was developed to address SemEval Task 7: "Determining Sentiment Intensity of English and Arabic Phrases", with focus on 'Arabic Phrases'. The goal of this task is to determine the degree to which some given term is associated with positive sentiment. The underlying premise behind the model that we have adopted is that determining the context (positive or negative) in which a term usually occurs can determine its strength. Since the focus is on Twitter terms, Twitter was used to collect tweets for each term for which a strength value was to be derived. An Arabic sentiment analyzer, was then used to assign a polarity to each of these tweets, thus defining their context. We then experimented with normalized point wise mutual information with and without linear regression to assign intensity scores to input terms. The output of the model that we've adopted ranked at two out of the three presented systems for this task with a **Kendall** score of 0.475.

1 Introduction

During the past few years, interest in sentiment analysis has surged. Sentiment lexicons are often an essential component for building sentiment analysis systems. Entries in sentiment lexicons can vary significantly in terms of how strongly they reflect positive or negative sentiment. For example, while both the terms "good" and "amazing" reflect a positive sentiment, the term "amazing" is stronger and more positive than "good". In this paper, we address early work that has been conducted to determine the intensity of Arabic twitter sentiment terms with respect to positive sentiment. Since the desired task is to determine the "positive strength" of a word, the results can be interpreted as follows: the closer the term score is to 1, the stronger it is as a positive indicator; the closer it is to 0, the stronger it is as a negative indicator. Terms that can be used in both positive and negative contexts will usually fall somewhere in the middle.

While this task was introduced in SemEval--2015 as Task 10- sub-task E (Rosenthal et al. 2015) for English terms, this is the first time that it has been introduced for Arabic terms. Addressing this task for Arabic has to take place in the absence of many resources that are available for English. In this work we make of use an Arabic sentiment analyzer (El-Beltagy et al. 2016) that was developed by our team as well as of a sentiment lexicon that was also developed within by same team and which is publicly available for research purposes (El-Beltagy 2016).

The main idea behind this work is to try to collect a representative number of tweets for each term for which a strength value is to be derived and to then classify those tweets in terms of polarity. This

Proceedings of SemEval-2016, pages 498–502,
San Diego, California, June 16-17, 2016. ©2016 Association for Computational Linguistics

classification would then be used for calculating the co-relation between positive sentiments and the original terms using normalized point wise mutual information (nPMI) (Bouma 2009). However, as will be detailed in section3.1, there were cases when tweets were not available for input terms, or when they were too few. A small sample of development data was also supplied with the task. We have basically used this data set to test the developed model and to make adjustments when needed.

The rest of this paper is organized as follows: section 2 briefly outlines related work; section 3 provides an overview of the developed model, section 4 presents the evaluation results and future directions, while section 5 concludes this paper.

2 Related Work

Because sentiment lexicons are an integral part of many sentiment analysis systems, many such lexicons have been developed for the English language. The most commonly used English lexicons include: SentiWordNet (Baccianella et al. 2010), Bing Liu's opinion lexicon(Liu 2010), and the MPQA subjectivity lexicon (Wilson et al. 2005), Recently, Twitter specific lexicons have also come into existence and are increasing being used. These include the Hashtag Sentiment Lexicon and the Sentiment140 Lexicon (Mohammad et al. 2013) (Kiritchenko et al. 2014). However despite the availability of many English lexicons, only a few include a sentiment score, with work on automatically assigning such a score only recently starting to attract attention. This particular research area was introduced as a subtask in SemEval-2015- task10. The top performing team for this sub-task, employed word embeddings to train a logistic regression model for assigning scores to sentiment terms (Astudillo et al. 2015). The second best performing team, used 6 different sentiment lexicons to score input terms (2 manually created, and 4 automatically created). Basically, input terms were compared against entries in the lexicons. If a term was found in a manually constructed lexicon, it was assigned a value of 1 or -1, depending on its polarity. If it was found in any of the automatically created lexicons, it was assigned the score found in those lexicons. If it was not found in any of the used lexicons, it was assigned a default value (Hamdan et al. 2015).

Arabic lexicons are much more scarce than their English counterparts, and are often translated versions of an existing English lexicon. An example of this is the Arabic translated version of the NRC word emotion association lexicon (EmoLex) (Mohammad & Turney 2013).

There have been some attempts to assign scores to Arabic lexicon terms. (El-Beltagy & Ali 2013) presented a method for semi-automatically building a sentiment lexicon as well as two different approaches for assigning scores to sentiment terms. The authors also demonstrated that the introduction of sentiment scores can increase the accuracy of sentiment analysis. (Eskander & Rambow 2015) constructed a sentiment lexicon by devising a matching algorithm that tries to match entries in the lexicon of an Arabic morphological analyzer to entries in SentiWordNet (Baccianella et al. 2010). When a match is found, a link is created between the lexicon entry and the matching entry in SentiWordNet and the scores of the matching term in SentiWordNet are assigned to that entry.

3 Model Overview

In order to assign strength scores to a given list of terms (input terms), a number of steps are carried out. These steps can be summarized as follows:

1. Collect tweets for input terms
2. Classify and index collected tweets
3. Calculate a score for each term

Each of the above steps is explained in the following subsections.

3.1 Data Collection

Since the goal of our work was to try to determine the correlation between a given term and positive or negative sentiments, we had to obtain a representative set of tweets for each term. We have chosen to retrieve 500 tweets for each term using Twitter's search API (Twitter 2016). There were cases however, when the search API was unable to retrieve this number of tweets and cases where no tweets were retrieved at all. It must also be noted that even though the flag that prevents retweets from being retrieved was set when invoking Twitter search, many tweets were in fact identical to other tweets in the tweet set. To eliminate those from the dataset, they were filtered out using the Jaccard similarity measure (Leskovec et al. 2014).

For cases when only very few tweets (less than 15) could be retrieved or when no tweets could be retrieved, an extra processing step was performed. In this step, a check was used to determine if the term in question was a hashtag. If it was, the hash symbol was removed from the term and so were any underscores. The resulting phrase was then used to query Twitter. If the term was not a hashtag or if the step just described also resulted in the retrieval of very few or no tweets, then the term was stemmed using a simple stemmer (El-Beltagy & Rafea 2011) and re-sent to Twitter as a query.

For the supplied 1166 test phrases in the Arabic set, 141 had to undergo these extra processing steps and 15 failed to return any tweets.

In total, approximately 249 K tweets were collected for deriving scores for the 1166 test phrases. This collection of tweets will henceforth be referred to as the twitter corpus.

3.2 Data Classification and Indexing

After carrying out the data collection step described in the previous section, each of the collected tweets was classified using the sentiment analyzer described in (El-Beltagy et al. 2016). The analyzer was built using a Complement Naïve Bayes classifier (Rennie et al. 2003) with a smoothing parameter of 1 and trained using 11,242 Arabic tweets of which 3759 were negative, 3725 were positive and 3758 were neutral. The prediction accuracy using 10 fold cross validation on this dataset was 79.4%. Complement Naïve Bayes was selected as a classifier based on the work presented in (Khalil et al. 2015).

Features related to the occurrence of positive and negative terms, were part of the feature-set used by the classifier. These features were determined using the NileULex sentiment lexicon (El-Beltagy 2016) which consists of 5953 single and compound MSA and Egyptian Arabic terms. The overlap between the input SemEval test set and NileULex was as follows: Out of the 1166 supplied test terms, 162 (13.8%) existed in the used lexicon, while 148(12.6%) hash-tagged terms, existed in the lexicon, but without the hashtag.

Some of the terms in both the development set and the test set, were negated. This was not really an issue when automatically assigning polarity labels, as the obtained label is one that it is relative to the entire phrase. What had to be handled however, were cases when the retrieved tweets included the negated form of a non-negated term. For example, when trying to assign a score to term “حلو”, a tweet with the following text:

“@mention امسحي كلامك عن□علاقة مع احمد مش حلو
ها□حكي في حقها”

would be classified as negative. In cases like this, negative labels were converted to positive ones and vice versa.

Each polarity classified tweet was then indexed using the Lucene(Apache 2011) search engine. The index was used to store the body of the tweet, the polarity class of the tweet, and the search term that was used to retrieve the tweet.

3.3 Term Scoring

Following the indexing and classification step, it was easy to retrieve information needed to calculate the *normalized pmi* (*npmi*) scores for each term relative the positive class. So for each term $term_i$ in the list for which strength values are to be derived, equation 1 was applied to calculate its *npmi* value relative to positive (pos) sentiment.

$$npmi(term_i, pos) = \frac{pmi(term_i,pos)}{-\log \mathrm{p}(term_i,\mathrm{pos})} \quad (1)$$

where

$$pmi(term_i, pos) = log\frac{p(term_i,pos)}{p(term_i)p(pos)} \quad (2)$$

$$p(term_i) = \frac{\text{\# of } term_i \text{in twitter corpus}}{size\ of\ twitter\ corpus} \quad (3)$$

$$p(pos) = \frac{\text{total number of positive tweets}}{size\ of\ twitter\ corpus} \quad (4)$$

$$p(\mathrm{term}_i, \mathrm{pos}) = \frac{\text{\# of pos term}_i\text{tweets}}{size\ of\ twitter\ corpus} \quad (5)$$

To calculate the probability of the occurrence of the i^th term in the test set ($p(term_i)$) in the twitter corpus, the number of tweets classified as neutral for this term ($term_i$) are subtracted from the total count of $term_i$ before applying equation 3. The total count of neutral tweets in the entire corpus is also subtracted from the size of the twitter corpus before applying equations 3 through 5.

Normalized point wise mutual information returns values in the range of [-1,1] where -1 indicates that a term would never occur with the positive class and 1, indicates that it always will. In our model, terms for which no tweets were retrieved received a *npmi* score of 0. The *nmpi* scores of terms that had to undergo extra processing as described in section 3.1 were penalized by multiplying their scores by a penalty factor <1. We have experimented with various factors using the development dataset, and ended up with 0.75 as a penalty factor.

We also experimented with two methods for re-scaling the *npmi* scores to values from 0 to 1. In the first model, we used the development dataset, which consisted of 200 terms, and then applied linear regression to map *npmi* scores to the gold standard scores. This linear regression model, was then applied on the scores of the supplied test set. In the second method, we applied simple re-scaling so that the scores would range from 0 to 1. When we experimented with both methods on the development dataset, simple re-scaling yielded slightly better results despite the fact that the regression model was built using that same set. Accordingly, the results that we've submitted, were the results obtained using simple rescaling rather the by applying linear regression.

4 Results and Discussion

Based on the results supplied by the organizers of the task, the presented approach ranked at number 2 as shown in **Table 1**.

		(official metric)	
Rank	**Team name**	**Kendall**	**Spearman**
1	iLab-Edinburgh	0.5362	0.67997
2	NileTMRG	0.47515	0.65763
3	LSIS	0.42431	0.58299

Table 1: Official results

We also ran the scoring script provided by the organizers on the results that were obtained using linear regression using the post submission test data with the gold labels. The results were as follows: **Kendall**: 0.47524, **Spearman:** 0.65756. The variation between these results and the submitted results are very minor.

By examining the results, it can be seen that the difference between our score and that achieved by the number 1 performer with respect to the Kendall metric is about 0.06. The difference in the Spearman metric is not quite as large. This difference as well as the score itself, suggests that there is still a need for improving this model.

The following points summarize some of the ideas that we plan on pursuing for improving this model:

1. Investigate alternative ways for handling the assignment of a score to terms for which no tweets were found.
2. Some terms had a higher percentage of duplicates than other in their tweet sets. Duplicate removal in cases like that resulted in those terms being under-represented. In the future, we intend to make sure that we get a pre-defined number of tweets for each term whenever possible.
3. We would like to repeat the experiment presented in this paper using tweets collected over a reasonable period of time, rather than in one go as we have done here. This should counter-act any time sensitivity issues for any given term.
4. We would like to examine the effect of replacing the sentiment analyzer for tweet labeling, with a method based on simple counting between the co-occurrence of the term for which we need to calculate a score, and known positive and negative terms.
5. Make better use of the existing lexicon as well as other available lexicons by experimenting with a similar approach to that presented in (Hamdan et al. 2015).

5 Conclusion

This paper presented the approach followed by NileTMRG for addressing Sem-Eval task 7, with respect to Arabic phrases. While the presented work shows promising results, further work is needed to improve its performance. In the previous section, we discussed several ideas that we believe might have a positive impact on the overall performance of the system.

Acknowledgments

The author wishes to thank the anonymous reviewers for their valuable feedback.

References

Apache, 2011. Lucene. Available at: http://lucene.apache.org/.

Astudillo, R.F. et al., 2015. INESC-ID: A Regression Model for Large Scale Twitter Sentiment Lexicon Induction. In *Proceedings of the 9th International Workshop on Semantic Evaluation (SemEval 2015)*. pp. 613–618. Available at: http://alt.qcri.org/semeval2015/cdrom/pdf/SemEval102.pdf.

Baccianella, S., Esuli, A. & Sebastiani, F., 2010. SentiWordNet 3.0: An Enhanced Lexical Resource for Sentiment Analysis and Opinion Mining. In *Proceedings of the Seventh International Conference on Language Resources and Evaluation (LREC'10)*. pp. 2200–2204.

Bouma, G., 2009. Normalized (Pointwise) Mutual Information in Collocation Extraction. In *the Biennial GSCL Conference*. pp. 31–40.

El-Beltagy, S.R. et al., 2016. Combining Lexical Features and a Supervised Learning Approach for Arabic Sentiment Analysis. In *CICLing 2016 - submitted*. Konya, Turkey.

El-Beltagy, S.R., 2016. NileULex: A Phrase and Word Level Sentiment Lexicon for Egyptian and Modern Standard Arabic. In *to appear in proceedings of LREC 2016*. Portorož , Slovenia.

El-Beltagy, S.R. & Ali, A., 2013. Open Issues in the Sentiment Analysis of Arabic Social Media : A Case Study. In *Proceedings of 9th the International Conference on Innovations and Information Technology (IIT2013)*. Al Ain, UAE.

El-Beltagy, S.R. & Rafea, A., 2011. An Accuracy Enhanced Light Stemmer for Arabic Text. *ACM Transactions on Speech and Language Processing*, 7(2), pp.2 – 23.

Eskander, R. & Rambow, O., 2015. SLSA: A Sentiment Lexicon for Standard Arabic. *Proceedings of the 2015 Conference on Empirical Methods in Natural Language Processing*, (September), pp.2545–2550. Available at: http://aclweb.org/anthology/D15-1304.

Hamdan, H., Bellot, P. & Bechet, F., 2015. lsislif: Feature extraction and label weighting for sentiment analysis in twitter. In *Proceedings of the 9th International Workshop on Semantic Evaluation*. pp. 568–573.

Khalil, T. et al., 2015. Which configuration works best? An experimental study on Supervised Arabic Twitter Sentiment Analysis. In *Proceedings of the First Conference on Arabic Computational Liguistics (ACLing 2015), co-located with CICLing 2015*. Cairo, Egypt, pp. 86–93.

Kiritchenko, S., Zhu, X. & Mohammad, S., 2014. Sentiment Analysis of Short Informal Texts. *Journal of Artificial Intelligence Research*, 50, pp.723–762.

Leskovec, J., Rajaraman, A. & Ullman, J.D., 2014. *Mining of Massive Datasets* 2 edition., Cambridge, UK: Cambridge University Press. Available at: http://ebooks.cambridge.org/ref/id/CBO9781139058452.

Liu, B., 2010. Sentiment Analysis and Subjectivity. In N. I. and F. J. Damerau, ed. *Handbook of Natural Language Processing, Second Edition*.

Mohammad, S. & Turney, P., 2013. Crowdsourcing a Word-Emotion Association Lexicon. *Computational Intelligence*, 29(3), pp.436–465.

Mohammad, S.M., Kiritchenko, S. & Zhu, X., 2013. NRC-Canada: Building the State-of-the-Art in Sentiment Analysis of Tweets. In *Proceedings of the seventh international workshop on Semantic Evaluation Exercises (SemEval-2013)*. Atlanta, Georgia, USA.

Rennie, J.D.M. et al., 2003. Tackling the Poor Assumptions of Naive Bayes Text Classifiers. *Proceedings of the Twentieth International Conference on Machine Learning (ICML)-2003)*, 20(1973), pp.616–623.

Rosenthal, S. et al., 2015. SemEval-2015 Task 10: Sentiment Analysis in Twitter. In *Proceedings of the 9th International Workshop on Semantic Evaluation (SemEval 2015)*. pp. 451–463. Available at: http://alt.qcri.org/semeval2015/cdrom/pdf/SemEval078.pdf.

Twitter, 2016. Twitter Search API. Available at: https://dev.twitter.com/rest/public/search.

Wilson, T., Wiebe, J. & Hoffmann, P., 2005. Recognizing contextual polarity in phrase-level sentiment analysis. In *Proceedings of Human Language Technology Conference and Conference on Empirical Methods in Natural Language Processing (HLT/EMNLP)*. Vancouver, Canada, pp. 347–354.