INESC-ID: A Regression Model for Large Scale Twitter Sentiment Lexicon Induction

Ramon F. Astudillo, Silvio Amir, Wang Ling, Bruno Martins†, Mário Silva, Isabel Trancoso
Instituto de Engenharia de Sistemas e Computadores Investigação e Desenvolvimento
Rua Alves Redol 9
Lisbon, Portugal
{ramon.astudillo, samir, wlin, mjs, isabel.trancoso}@inesc-id.pt
†bruno.g.martins@tecnico.ulisboa.pt

Abstract

We present the approach followed by INESC-ID in the SemEval 2015 Twitter Sentiment Analysis challenge, subtask E. The goal was to determine the strength of the association of Twitter terms with positive sentiment. Using two labeled lexicons, we trained a regression model to predict the sentiment polarity and intensity of words and phrases. Terms were represented as word embeddings induced in an unsupervised fashion from a corpus of tweets. Our system attained the top ranking submission, attesting the general adequacy of the proposed approach.

1 Introduction

Sentiment lexicons are one of the key resources for the automatic analysis of opinions, emotive and subjective text (Liu, 2012). They compile words annotated with their *prior polarity* of sentiment, regardless of the context. For instance, words such as *beautiful* or *amazing* tend to express a positive sentiment, whereas words like *boring* or *ugly* are considered negative. Most sentiment analysis systems use either *word count* methods, based on sentiment lexicons, or rely on text classifiers. In the former, the polarity of a message is estimated by computing the ratio of (positive and negative) sentiment bearing words. Despite its simplicity, this method has been widely used (O'Connor et al., 2010; Bollen and Mao, 2011; Mitchell et al., 2013). Even more sophisticated systems, based on supervised classification, can be greatly improved with features derived from lexicons (Kiritchenko et al., 2014). However, manually created sentiment lexicons consist of few carefully selected words. Consequently, they fail to capture the use of non-conventional word spelling and slang, commonly found in social media.

This problem motivated the creation of a task in the SemEval 2015 Twitter Sentiment Analysis challenge. This task (subtask E), intended to evaluate automatic methods of generating Twitter specific sentiment lexicons. Given a set of words or phrases, the goal was to assign a score between 0 and 1, reflecting the intensity and polarity of sentiment these terms express. Participants were asked to submit a list, with the candidate terms ranked according to sentiment score. This list was then compared to a ranked list obtained from human annotations and the submissions were evaluated using the Kendall (1938) Tau rank correlation metric.

In this paper, we describe a system developed for this challenge, based on a novel method to create large scale, domain-specific sentiment lexicons. The task is addressed as a regression problem, in which terms are represented as word embeddings, induced from a corpus of 52 million *tweets*. Then, using manually annotated lexicons, a regression model was trained to predict the polarity and intensity of sentiment of any word or phrase from that corpus. We found this approach to be effective for the proposed problem.

The rest of the paper proceeds as follows: we review the work related to lexicon expansion in Section 2 and describe the methods used to derive word embeddings in Section 3. Our approach and the experimental results are presented in Sections 5 and 6, respectively. We conclude in Section 7.

Proceedings of the 9th International Workshop on Semantic Evaluation (SemEval 2015), pages 613–618,
Denver, Colorado, June 4-5, 2015. ©2015 Association for Computational Linguistics

2 Related Work

Most of the literature on automatic lexicon expansion consists of dictionary-based or corpora-based approaches. In the former, the main idea is to use a dictionary, such as *WordNet*, to extract semantic relations between words. Kim and Hovy (2006) simply assign the same polarity to synonyms and the opposite polarity to antonyms, of known words. Others, create a graph from the semantic relationships, to find new sentiment words and their polarity. Using the seed words, new terms are classified using a distance measure (Kamps et al., 2004), or propagating the labels along the edges of the graph (Rao and Ravichandran, 2009). However, given that dictionaries mostly describe conventional language, these methods are unsuited for social media.

Corpora based approaches follow the assumption that the polarity of new words can be inferred from co-occurrence patterns with known words. Hatzivassiloglou and McKeown (1997) discovered new polar adjectives by looking at conjunctions found in a corpus. The adjectives connected with *and* got the same polarity, whereas adjectives connected with *but* were assigned opposing polarities. Turney and Littman (2003) created two small sets of prototypical polar words, one containing positive and another containing negative examples. The polarity of a new term was computed using the point-wise mutual information between that word and each of the prototypical sets (Lin, 1998). The same method was used by Kiritchenko et al. (2014), to create large scale sentiment lexicons for Twitter.

A recently proposed alternative is to learn word embeddings specific for Twitter sentiment analysis, using distant supervision (Tang et al., 2014). The resulting features are then used in a supervised classifier to predict the polarity of phrases. This work is the most related to our approach, but it differs in the sense that we use general word embeddings, learned from unlabeled data, and predict both polarity and intensity of sentiment.

3 Unsupervised Word Embeddings

In recent years, several models have been proposed, to derive *word embeddings* from large corpora. These are essentially, dense vector representations that implicitly capture syntactic and semantic properties of words (Collobert et al., 2011; Mikolov et al., 2013a; Pennington et al., 2014). Moreover, a notion of *semantic similarity*, as well as other linguistic regularities seem to be encoded in the embedding space (Mikolov et al., 2013b). In `word2vec`, Mikolov et al. (2013a) induce word vectors with two simple neural network architectures, CBOW and skip-gram. These models estimate the optimal word embeddings by maximizing the probability that, words within a given window size are predicted correctly.

Skip-gram and Structured Skip-gram

Central to the **skip-gram** is a log-linear model of word prediction. Given the i-th word from a sentence w_i, the skip-gram estimates the probability of each word at a distance p from w_i as:

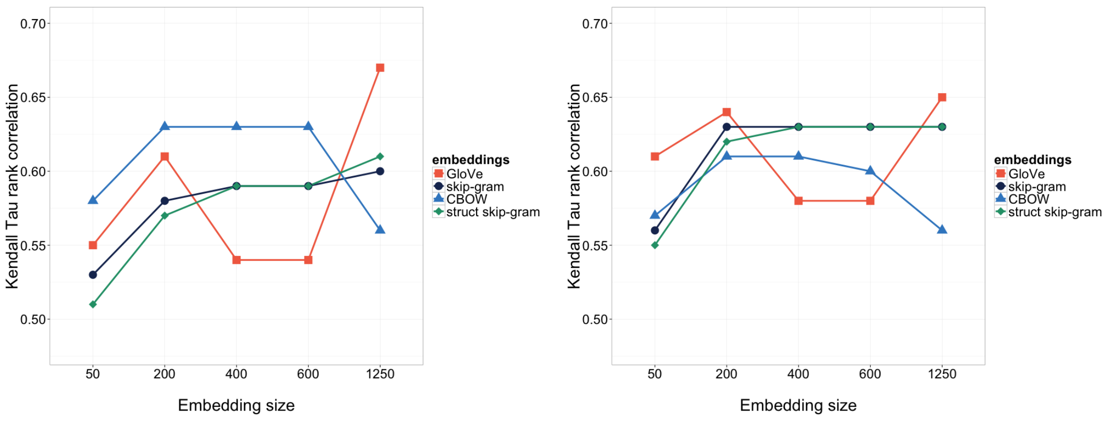
$$p(w_{i+p}|w_i; \mathbf{C}_p, \mathbf{E}) \propto \exp\left(\mathbf{C}_p \cdot \mathbf{E} \cdot \mathbf{w}_i\right) \quad (1)$$

Here, $\mathbf{w}_i \in \{1, 0\}^{v \times 1}$ is a one-hot representation of the word, i.e., a sparse column vector of the size of the vocabulary v with a 1 on the position corresponding to that word. The model is parametrized by two matrices: $\mathbf{E} \in \mathbb{R}^{e \times v}$ is the embedding matrix, transforming the one-hot sparse representation into a compact real valued space of size e; $\mathbf{C}_p \in \mathbb{R}^{v \times e}$ is a matrix mapping the real-valued representation to a vector with the size of the vocabulary v. A distribution over all possible words is then attained by exponentiating and normalizing over the v possible options. In practice, due to the large value of v, various techniques are used to avoid having to normalize over the whole vocabulary (Mikolov et al., 2013a). In the particular case of the **structured skip-gram** model, the matrix $\mathbf{C}_p$ depends only of the relative position between words p (Ling et al., 2015).

After training, the low dimensional embedding $\mathbf{E} \cdot \mathbf{w}_i \in \mathbb{R}^{e \times 1}$ encapsulates the information about each word and its surrounding contexts.

CBOW

The CBOW model defines a different objective function, that predicts a word at position i given the window of context $i - d$, where d is the size of the context window. The probability of the word w_i is



(a) Phrases as the **sum** of embeddings

(b) Phrases as the **mean** of embeddings

Figure 1: Performance of the different embeddings and phrase representations, as function of vector size.

defined as:

$$p(w_i|w_{i-d}, ..., w_{i+d}; \mathbf{C}, \mathbf{E}) \propto \exp(\mathbf{C} \cdot \mathbf{S}_{i-d}^{i+d}) \quad (2)$$

where $\mathbf{S}_{i-d}^{i+d}$ is the point wise sum of the embeddings of all context words starting at $\mathbf{E} \cdot w_{i-d}$ to $\mathbf{E} \cdot w_{i+d}$, excluding the index w_i, and once again $\mathbf{C} \in \mathbb{R}^{e \times v}$ is a matrix mapping the embedding space into the output vocabulary space v.

GloVe

The models discussed above rely on different assumptions about the relations between words within a context window. The Global Vector model, referred as GloVe (Pennington et al., 2014), combines this approach with ideas drawn from matrix factorization methods, such as LSA (Deerwester et al., 1990). The embeddings are derived with an objective function that combines context window information, with corpus statistics computed efficiently from a global term co-occurrence matrix.

4 Labeled Data

The evaluation of the shared task was performed on a labeled test set, consisting of 1315 words and phrases. To support the development of the systems, the organizers released a *trial* set with 200 examples. The terms are representative of the informal style of Twitter text, containing hashtags, slang, abbreviations and misspelled words. Negated expressions were also included. We show a sample of the words and phrases in Table 1. For more details on these datasets, see (Kiritchenko et al., 2014).

Given the small size of the trial set, we used an additional labeled lexicon: the Language Assessment by Mechanical Turk (LabMT) lexicon (Dodds et al., 2011). It consists of 10,000 words collected from different sources. Words were rated on a scale of 1 (sad) to 9 (happy), by users of Amazon's Mechanical Turk service, resulting in a measure of average happiness for each given word. Note that LabMT contains annotations for *happiness* but our goal is to label words in terms of *sentiment polarity*. We rely on the fact that some emotions are correlated with sentiment, namely, joy/happiness are associated with positivity, while sadness/disgust relate to negativity (Liu, 2012).

This complementary dataset was used for two purposes: first, as the development set to evaluate and tune our system, and second, as additional training data for the candidate submission.

Type	**Sample words**
words	*sweetest, giggle, sleazy, broken*
slang	*bday, lmao, kewl, pics*
negations	*can't cope, don't think, no probs*
interjections	*weee, yays, woooo, eww*
emphasized	*gooooood, loveeee, cuteeee, excitedddd*
hashtags	*#gorgeous, #smelly, #fake, #classless*
multiword hashtag	*#goodvibes, #everyonelikesitbutme*
emoticons	*:o): -.- :') <33*

Table 1: A sample of the different types of terms.

5 Proposed Approach

We addressed the task of inducing large scale sentiment lexicons for Twitter as a regression problem. Each term $\mathbf{w}_i$ was represented with an embedding $\mathbf{E} \cdot \mathbf{w}_i \in \mathbb{R}^{e \times 1}$, with $e \in \{50, 200, 400, 600, 1250^1\}$ as discussed in Section 3. Then, the manually annotated lexicons were used to train a model that, given a new term $\mathbf{w}_j$, predicts a score $y \in [0, 1]$ reflecting the polarity and intensity of sentiment it conveys.

Note that the embeddings represent words, so to deal with phrases we leveraged on the compositional properties of word vectors (Mikolov et al., 2013b). Given that algebraic operations in the embedding space preserve meaning, we represented phrases as the sum or mean of individual word vectors.

5.1 Learning the Word Embeddings

The first step of our approach, requires a corpus of tweets to support the unsupervised learning of the embedding matrix $\mathbf{E}$. We resorted to the corpus of 52 million tweets used by Owoputi et al. (2013) and the tokenizer described in the same work.

The CBOW and skip-gram embeddings were induced using the `word2vec`[2] tool, while we used our own implementation of the structured skip-gram. The default values in `word2vec` were employed for most of the parameters, but we set the negative sampling rate to 25 words (Goldberg and Levy, 2014). For the GloVe model, we used the available implementation[3] with the default parameters. In all the models, words occurring less than 100 times in the corpus were discarded, resulting in a vocabulary of around 210,000 tokens.

Finally, embeddings of different sizes were built, with 50, 200, 400 and 600 dimensions.

Hyperparameter Optimization and Model Selection

Regarding the choice of learning algorithm, several linear regression models were considered: least squares and regularized variants, namely, the *lasso*, *ridge* and *elastic net* regressors. We also experimented with *Support Vectors Regression (SVR)* using non-linear kernels, namely, polynomial, sigmoid and Radial Basis Function (RBF). Most of these models have hyperparameters, thus the combination of possible algorithms and parameters represents a huge configuration space. A brute force approach to find the optimal model would be cumbersome and time consuming. Instead, for each parameter, we defined meaningful distributions and ranges of values. Then, a hyperparameter optimization algorithm was used to find the best combination of model and parameters, by sampling from the specified configuration pool. The *Tree of Parzen Estimators* algorithm, as implemented in `HyperOpt`[4], was used (Bergstra et al., 2013).

6 Experiments

Learning word embeddings from large corpora allowed us to derive representations for a considerable number of words. Thus, we were able to find embeddings for 94% of the candidate terms. Using simple normalization steps, we could find embeddings for the remaining terms. However, we found that this improvement in recall had almost no impact in the performance of the system.

After mapping terms to their respective embeddings, we performed experiments to find the best regression model and respective hyperparameters. For this purpose, the LabMT lexicon was employed as the development set and the trial data as a validation set, against which different configurations were evaluated. After 1000 trials, the SVR model with RBF kernel was selected. Finally, we performed detailed experiments to compare word embedding models and vectors of different dimensions.

6.1 Submitted System

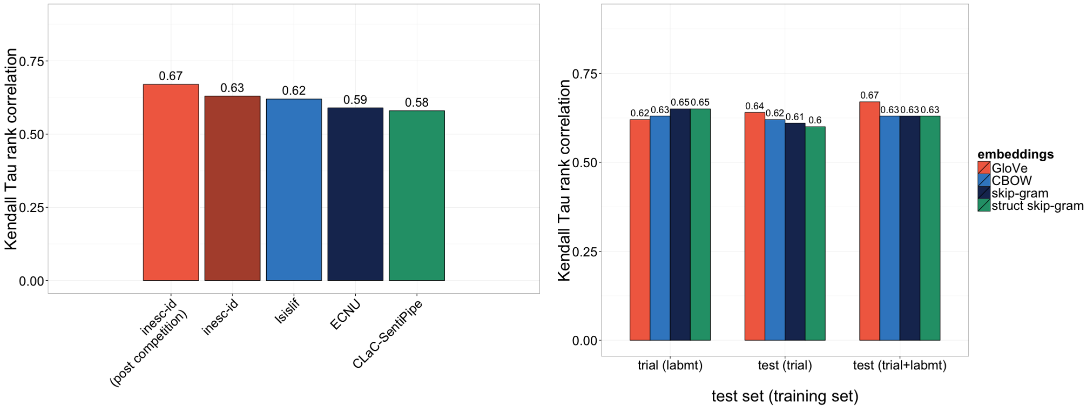
The evaluation on the trial data indicated that several configurations of embedding model and size could achieve the optimal results. Therefore, our candidate system was based on structured skip-gram embeddings with 600 dimensions, and SVR with RBF kernel. The hyperparameters were set to $C = 50$, $\epsilon = 0.05$ and $\gamma = 0.01$ and the system was trained using the trial data and the LabMT lexicon.

[1] corresponds to the concatenation of all the embeddings

[2] `https://code.google.com/p/word2vec/`

[3] `http://nlp.stanford.edu/projects/GloVe/`

[4] `http://hyperopt.github.io/hyperopt/`



(a) Results of the top 4 ranking systems

(b) Comparing word embedding models under various training and test data regimes

Figure 2: Evaluation of the INESC-ID system.

6.2 Results

The experiments showed that all the word embeddings have comparable capabilities. In Figure 1, we compare the results of different embeddings with the same regression model. Regarding the representation of phrases, the skip-gram and structured skip-gram embeddings performed better when averaged. However, the GloVe and CBOW seemed to be more effective when summing the individual word vectors. These results were consistent across all the experiments. In terms of embedding size, we observed that smaller vectors tend to perform worse and, in general, concatenating vectors of different dimensionality improved performance. The CBOW representations were the only exception. This suggests that embeddings of different size capture different aspects of words.

Our final method, attained the highest ranking result of the competition, with 0.63 rank correlation. Figure 2a shows the results of the top 4 submissions to SemEval. Further experiments were conducted after the release of the test set labels. We found that the concatenation of GloVe embeddings outperforms our previous choice of features on the test set. Surprisingly, these embeddings obtained the worst results on the trial data, but are much better than the others in the test set, achieving a rank correlation of 0.67. At this point, it is still not clear why this is the case.

Figure 2b shows the performance of each embedding model, under different combinations of training and test data. We can see that the proposed approach is effective, and our models outperform the other systems with as few as 200 training examples.

7 Conclusions

We described the approach followed by INESC-ID for subtask E of SemEval 2015 Twitter Sentiment Analysis challenge. This work presents the first steps towards a general method to extract large-scale lexicons with fine-grained annotations from Twitter data. Although the results are encouraging, further investigation is required to shed light on some unexpected outcomes (e.g., the inconsistent behavior of the GloVe features on the trial and test sets). It should nonetheless be noted that, given the small size of the labeled sets, it is hard to draw definitive conclusions about the soundness of any method. Furthermore, the merit of a sentiment lexicon should be assessed in terms of its impact on the performance of concrete sentiment analysis applications.

Acknowledgements

This work was partially supported by Fundação para a Ciência e Tecnologia (FCT), through contracts Pest-OE/EEI/LA0021/2013, EXCL/EEI-ESS/0257/2012 (DataStorm), grant number SFRH/BPD/68428/2010 and Ph.D. scholarship SFRH/BD/89020/2012.

References

James Bergstra, Daniel Yamins, and David Cox. 2013. Making a science of model search: Hyperparameter optimization in hundreds of dimensions for vision architectures. In *Proceedings of the 30th International Conference on Machine Learning*, pages 115–123.

Johan Bollen and Huina Mao. 2011. Twitter mood as a stock market predictor. *Computer*, 44:91–94.

Ronan Collobert, Jason Weston, Léon Bottou, Michael Karlen, Koray Kavukcuoglu, and Pavel Kuksa. 2011. Natural language processing (almost) from scratch. *The Journal of Machine Learning Research*, 12:2493–2537.

Scott C. Deerwester, Susan T Dumais, Thomas K. Landauer, George W. Furnas, and Richard A. Harshman. 1990. Indexing by latent semantic analysis. *JAsIs*, 41:391–407.

Peter Sheridan Dodds, Kameron Decker 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.

Yoav Goldberg and Omer Levy. 2014. word2vec explained: deriving mikolov et al.'s negative-sampling word-embedding method. *arXiv preprint arXiv:1402.3722*.

Vasileios Hatzivassiloglou and Kathleen R McKeown. 1997. Predicting the semantic orientation of adjectives. In *Proceedings of the 8th Conference of the European Chapter of the Association for Computational Linguistics*, pages 174–181.

Jaap Kamps, Maarten Marx, Robert J. Mokken, and Maarten de Rijke. 2004. Using wordnet to measure semantic orientations of adjectives. In *Proceedings of 4th International Conference on Language Resources and Evaluation, Vol IV,*, pages 1115–1118.

Maurice G Kendall. 1938. A new measure of rank correlation. *Biometrika*, pages 81–93.

Soo-Min Kim and Eduard Hovy. 2006. Identifying and analyzing judgment opinions. In *Proceedings of the Human Language Technology Conference of the NAACL, Main Conference*, pages 200–207.

Svetlana Kiritchenko, Xiaodan Zhu, and Saif M Mohammad. 2014. Sentiment analysis of short informal texts. *Journal of Artificial Intelligence Research*, pages 723–762.

Dekang Lin. 1998. An information-theoretic definition of similarity. In *Proceedings of the 15th International Conference on Machine Learning*, volume 98, pages 296–304.

Wang Ling, Chris Dyer, Alan Black, and Isabel Trancoso. 2015. Two/too simple adaptations of word2vec for syntax problems. In *Proceedings of the 2015 Conference of the North American Chapter of the Association for Computational Linguistics: Human Language Technologies*.

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

Tomas Mikolov, Kai Chen, Greg Corrado, and Jeffrey Dean. 2013a. Efficient estimation of word representations in vector space. In *Workshop at the International Conference on Learning Representations*.

Tomas Mikolov, Ilya Sutskever, Kai Chen, Greg Corrado, and Jeffrey Dean. 2013b. Distributed representations of words and phrases and their compositionality. In *Proceedings of the 27th Annual Conference on Neural Information Processing Systems*.

Lewis Mitchell, Kameron Decker Harris, Morgan R Frank, Peter Sheridan Dodds, and Christopher M Danforth. 2013. The geography of happiness: connecting twitter sentiment and expression, demographics, and objective characteristics of place. *PLoS ONE*, 8(5).

Brendan O'Connor, Ramnath Balasubramanyan, Bryan R Routledge, and Noah A Smith. 2010. From tweets to polls: Linking text sentiment to public opinion time series. In *Proceedings of the 4th International AAAI Conference on Weblogs and Social Media*.

Olutobi Owoputi, Brendan O'Connor, Chris Dyer, Kevin Gimpel, Nathan Schneider, and Noah A Smith. 2013. Improved part-of-speech tagging for online conversational text with word clusters. In *Proceedings of the 2013 Conference of the North American Chapter of the Association for Computational Linguistics: Human Language Technologies*, pages 380–390.

Jeffrey Pennington, Richard Socher, and Christopher D Manning. 2014. Glove: Global vectors for word representation. *Proceedings of the 2014 Empiricial Methods in Natural Language Processing*, 12.

Delip Rao and Deepak Ravichandran. 2009. Semi-supervised polarity lexicon induction. In *Proceedings of the 12th Conference of the European Chapter of the Association for Computational Linguistics*, pages 675–682.

Duyu Tang, Furu Wei, Bing Qin, Ming Zhou, and Ting Liu. 2014. Building large-scale twitter-specific sentiment lexicon : A representation learning approach. In *Proceedings of the 25th International Conference on Computational Linguistics*, pages 172–182.

Peter D Turney and Michael L Littman. 2003. Measuring praise and criticism: Inference of semantic orientation from association. *ACM Transactions on Information Systems (TOIS)*, 21(4):315–346.