AMRITA_CEN@SemEval-2015: Paraphrase Detection for Twitter using Unsupervised Feature Learning with Recursive Autoencoders

Mahalakshmi Shanmuga Sundaram, Anand Kumar Madasamy and Soman Kotti Padannayil
Center for Excellence in Computational Engineering and Networking
Amrita Vishwa Vidyapeetham
Coimbatore, India
mahalakshmisklu@gmail.com
m_anandkumar@cb.amrita.edu
kp_soman@amrita.edu

Abstract

We explore using recursive autoencoders for SemEval 2015 Task 1: Paraphrase and Semantic Similarity in Twitter. Our paraphrase detection system makes use of phrase-structure parse tree embeddings that are then provided as input to a conventional supervised classification model. We achieve an F1 score of 0.45 on paraphrase identification and a Pearson correlation of 0.303 on computing semantic similarity.

1 Introduction

The process of rewriting text with a different choice of words or using a different sentence structure while preserving meaning is called paraphrasing. Identifying paraphrases can be a difficult task owing to the fact that evaluating surface level similarity is often not enough, but rather systems must take into account the underlying semantics of the content being assessed.

Paraphrasing and paraphrase detection are important and challenging tasks, which find their application in various subfields of Natural Language Processing (NLP) such as information retrieval, question answering (Erwin and Emiel, 2005), plagiarism detection (Paul Clough et al., 2002), text summarization and evaluation of machine translation (Chris Callison Burch, 2008).

We explore using recursive autoencoders for paraphrase detection and similarity scoring as a part of SemEval 2015 Task 1: Paraphrase and Semantic Similarity in Twitter. Twitter is an online social networking service with millions of users who casually converse about diverse topics in a continuous and contemporaneous manner (Wei Xu et al., 2014; Wei Xu et al., 2015). Table 1 gives an example of real tweets, some of which are paraphrases of each other. The very casual style of the Twitter corpus makes it more challenging to work with for many NLP tools. We use vector space embeddings, in part, since they are relatively good at dealing with noisy data.

2 Related Work

Socher et al. (2011) explored using recursive autoencoders (RAEs) and dynamic pooling for paraphrase detection. They parse each sentence within a pair, compute embeddings for each node in the parse trees, and then construct a similarity matrix comparing the embedding vectors for all nodes within the two parse trees. Using dynamic pooling, they convert the variable size similarity matrix for each sentence pair to a matrix of fixed size. The resulting fixed size matrix is then given to a softmax classifier to detect whether the sentences are paraphrases.

3 A Deep Learning System

The architecture of our system is depicted in Figure 1. The raw Twitter corpus is preprocessed using a phrase-structure parser. The resulting parse trees are then used to train an unfolding RAE model. This model provides us with embedding vectors that are then used to compute the similarity between every node in the parse trees associated with a sentence pair. A similarity matrix is populated with the node-to-node similarity scores as measured by the Euclidean distance beween the node embedding vectors. The size of the similarity matrix depends on

Proceedings of the 9th International Workshop on Semantic Evaluation (SemEval 2015), pages 45–50,
Denver, Colorado, June 4-5, 2015.

Sentence 1	**Sentence 2**	**Paraphrase or Not**
AAP is in the Adidas commercial	AAP in that Adidas Commercial lol	Paraphrase
That amber alert was getting annoying	Why do I get amber alerts tho	Not paraphrase
I am so watching Cinderella right now	Im so watching Cinderella right now	Paraphrase
That shot counted by Bayless	Bayless just RAN for it	Not Paraphrase
Damon EJ 1st Qb off the board	if EJ is the 1st QB off the board	Paraphrase

Table 1: Sample tweets from SemEval 2015 Twitter Paraphrase Corpus.

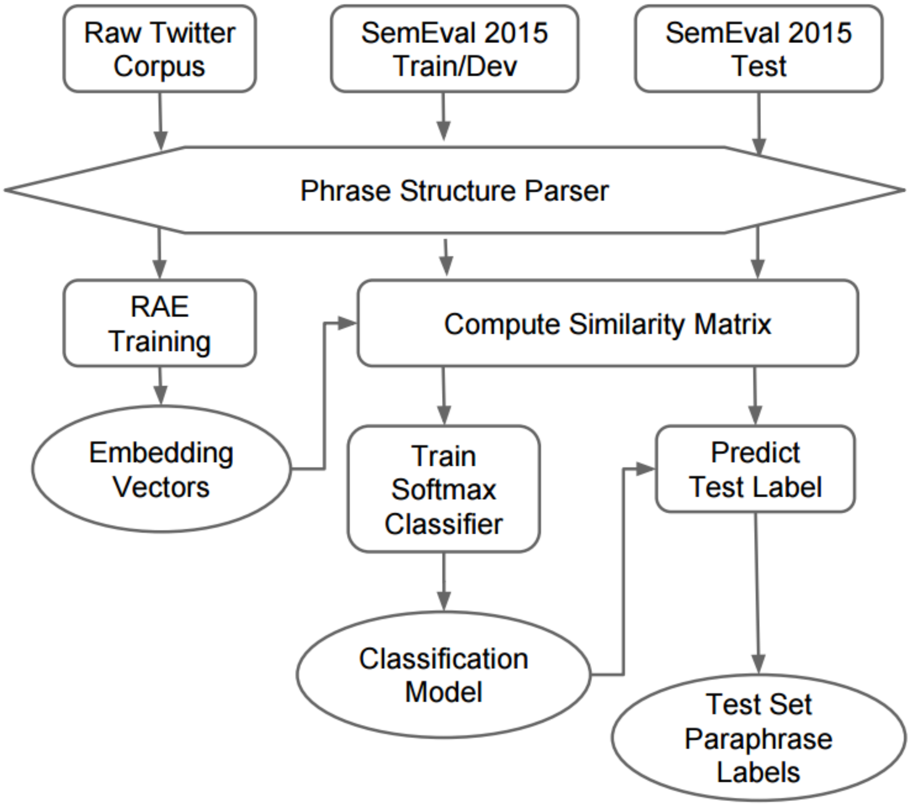

Figure 1: System architecture: The unfolding recursive autoencoder computes phrase embedding vectors for each node in a parse tree. For a pair of sentences being evaluated, the distances between all the nodes in the paired parse trees are computed and fill a variable sized similarity matrix. Dynamic pooling is used to convert the variable size similarity matrix to fixed size matrix. The fixed size similarity matrix is given to a softmax classifier to detect both whether the paired sentences are paraphrases and for paraphrase similarity scoring.

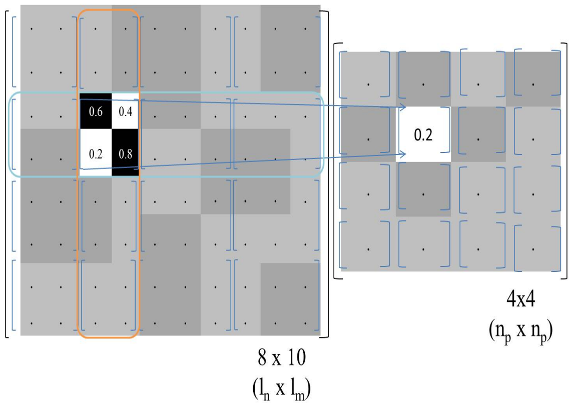

Figure 2: Dynamic pooling: The original variable sized matrix is partitioned into an $n_p \times n_p$ grid of blocks of approximately equivalent size. We use *min-pooling* as the aggregation operation, whereby the values of the cells in the fixed size $n_p \times n_p$ matrix are assigned to the minimum value of the corresponding partition in the original matrix.

the number of nodes in the parse trees being compared. This variable size similarity matrix is converted to a fixed size matrix using Dynamic Pooling (Socher et.al, 2011). Dynamic pooling partitions the rows and columns of similarity matrix into n_p approximately equivalent segments which creates an $n_p \times n_p$ grid. As depicted in Figure 2, the individual cells in the fixed size $n_p \times n_p$ matrix are assign to the minimum values of their corresponding partitions in the original matrix. The resulting fixed size matrix is then used to train a softmax classifier to perform the actual paraphrase detection and pairwise similarity scoring tasks. To classify a pair of new sentences, the sentences are first parsed. Using the parse trees, the embedding vectors for each sentence are constructed and used to populate a node-to-node similarity matrix. This matrix is converted to a fixed size using dynamic pooling and passed to the softmax classification model.

3.1 Unfolding Recursive Autoencoders (RAEs)

The architecture of our unfolding RAEs is illustrated in Figure 2. The main difference between standard RAEs and unfolding RAEs is that standard RAEs are only directly trained to have each node reconstruct its immediate children. Unfolding RAEs differ in that the training objective assess not only how well the representation of each node reconstructs it's immediate children, but rather how well the node's representation reconstructs the entire parse tree fragment rooted at the current node.

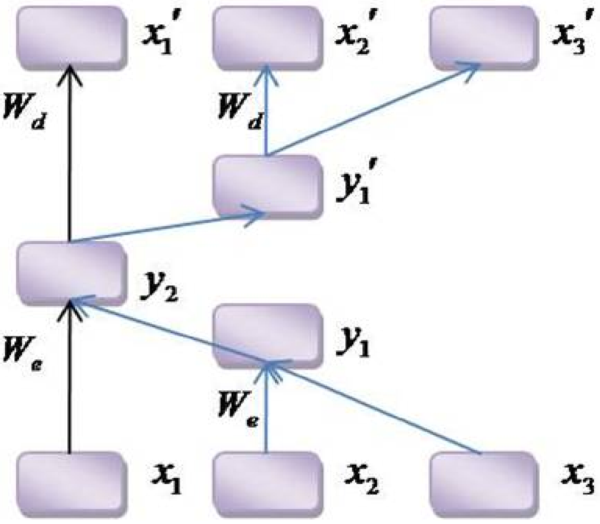

Figure 3: Architecture of unfolding RAEs. Using unfolding RAEs, the embedding vector associated with each node in a parse tree is trained to reconstruct the whole parse tree fragment rooted at the current node.

4 Experimental Results

We use a general domain parsing model distributed with the Stanford Parser, englishPCFG v1.6.9 (Klein and Manning, 2003). Prior to training the RAE vectors, we pre-trained word embedding vectors for use as the word level representations (Ronan and Jason, 2008). The hyperparameter values used for our system are as follows: (1) the size of the pooling matrix n_p = 13; (2) the regularization for the softmax classifier c = 0.05; (3) Both the RAE and word embeddings are 100-dimensional vectors.

4.1 Data Set Details

Our SemEval task provided the PIT-2015 Twitter Paraphrase corpus for training and system development (Wei Xu, 2014; Wei Xu et al., 2014; Wei Xu et al., 2015). The corpus contains a training set with 13,063 sentence pairs, a development set with 4,727 sentence pairs, and a test set with 972 sentence pairs. Table 2 shows the label distribution statistics for this corpus. This data set is distinct from the data used

Category	Paraphrase Sentence pair	Non-Paraphrase Sentence pair	Debatable Sentence pair	Total
Training	3,996	7,534	1,533	13,063
Development	1,470	2,672	585	4,727
Testing	175	663	134	972

Table 2: Statistics of PIT-2015 Twitter Paraphrase Corpus.

Twitter Corpus	Training	Testing/ Development	Precision	Recall	F1 Measure
50,000	13,063	4,727	0.51	0.48	0.49
80,000	13,063	4,727	0.65	0.37	0.51
95,000	13,063	4,727	0.77	0.35	0.56

Table 3: PIT-2015 dev set performance using varying amounts of training data.

in other work on paraphrasing in the following ways: (1) it contains sentences that are colloquial and opinionated; (2) it contains paraphrases that are lexically diverse; and (3) it contains many sentences that are lexically similar but semantically dissimilar (Wei Xu et al., 2015).

The training and development data was jointly collected from 500+ trending topics and then randomly split into the final training and development sets. The test data was drawn from 20 randomly sampled Twitter trending topics. Labels were collected by having each sentence pair annotated by 5 different crowdsourced workers.

4.2 Evaluation and Discussion

For the unsupervised unfolding RAE training, we experimented with using subsets of different sized Twitter corpora of 50,000, 80,000 and 95,000 sentences to evaluate the proposed system. Using PIT-2015, we trained using tweets from the training set and evaluated the resulting series of systems on the dev set (Wei Xu et al., 2015). For supervised training, we used the training set from PIT-2015. For training the unsupervised unfolding RAE vectors, we collected additional data using the Twitter Developer API. As shown in Table 3, we found that increasing the size of the data set used to train the RAE embeddings leads to strong gains in system performance.[1] Notice that as the amount of data used to train the RAE vectors increases, the precision value for paraphrase detection increases significantly while the recall value is actually falling.

Metrics Type	Accuracy
maxF1	0.457
mPrecision	0.543
mRecall	0.394
Pearson	0.303

Table 4: Results from the SemEval-2015.

The official evaluation metrics for SemEval-2015 Task 1 are F1-score for paraphrase identification and Pearson correlation for the semantic similarity scores. The performance of our system on the shared task evaluation data using these metrics is presented in Table 4.

5 Conclusion and Future Work

We participated in SemEval 2015 Task 1: Paraphrase and Semantic Similarity in Twitter using a system architecture motivated by the success of prior work on using RAE for paraphrase detection (Socher et al. 2011). We find that the performance of the system receives a sizable boost with the addition of a moderate amount of unsupervised RAE training data.

In future work, we plan to try to improve performance by first normalizing the Twitter data prior to parsing. Given the mismatch between general domain English data and tweets, parse accuracy would have likely been improved by performing a preprocessing step that normalized the tweets prior to

[1] Due to time constraints we did not explore using more than 95,000 sentences to train our embedding model.

giving them to the parser (Juri Ganitkevitch et al., 2013; Brendan O Connor et al., 2010). This could lead to improved downstream paraphrase detection and similarity scoring. We would also like to explore using new learning algorithms for the final paraphrase classification as well as alternative mechanisms of constructing the sentence level embedding vectors.

Acknowledgments

In this work, we would like to convey our sincere gratitude and special thanks towards Wei Xu, organizer of SemEval PIT 2015, who helped us in the training and development data set and to evaluate our system results. We would like again to convey our sincere gratitude towards Daniel Cer, who encouraged and motivated us throughout the final submission. And we would convey our sincere thanks to all the organizers of SemEval 2015.

References

Bill Dolan., Chris Quirk and Chris Brockett. 2004. *Unsupervised construction of large paraphrase corpora: Exploiting massively parallel news sources.* Proceedings of the 20th international conference on Computational Linguistics (pp. 350).

Brendan O Connor., Michel Krieger and David Ahn. 2010. *TweetMotif: Exploratory Search and Topic Summarization for Twitter.*

Chris Callison Burch. 2008. *Syntactic constraints on paraphrases extracted from parallel corpora.* Proceedings of the Conference on Empirical Methods in Natural Language Processing (pp. 196-205).

Dan Klein and Christopher D. Manning. 2003. *Accurate unlexicalized parsing.* Proceedings of the 41st Annual Meeting on Association for Computational Linguistics-Volume 1, pp. 423-430.

Duyu Tang., Furu Wei., Bing Qin., Ting Liu and Ming Zhou. 2014. *Coooolll: A Deep Learning System for Twitter Sentiment Classification.* Proceedings of the 8th International Workshop on Semantic Evaluation (SemEval-2014). (pp. 208-212).

Eric Huang 2011. *Paraphrase Detection Using Recursive Autoencoder.*

Erwin Marsi and Emiel Krahmer 2005. *Explorations in sentence fusion.* Proceedings of the European Workshop on Natural Language Generation (pp. 109-117).

Fabio Massimo Zanzotto., Marco Pennacchiotti and Kostas Tsioutsiouliklis. 2011. *Linguistic redundancy in twitter.* In Proceedings of the Conference on Empirical Methods in Natural Language Processing (pp. 659-669).

Juri Ganitkevitch., Benjamin Van Durme and Chris Callison-Burch. 2013. *PPDB: The Paraphrase Database.* In HLT-NAACL (pp. 758-764).

Leon Derczynski., Alan Ritter., Sam Clark and Kalina Bontcheva. 2013. *Twitter Part-of-Speech Tagging for All: Overcoming Sparse and Noisy Data.* In RANLP (pp. 198-206).

Microsoft research paraphrase corpus. Accessed on September-2014. *http://research.microsoft.com/en-us/.*

Nitin Madnan and Bonnie J. Dorr. 2010. *Generating phrasal and sentential paraphrases: A survey of data-driven methods.* Computational Linguistics, 36(3), (pp. 341-387).

Paul Clough., Robert Gaizauskas., Scott SL Piao and Yorick Wilks. 2002. *Meter: Measuring text reuse.* Proceedings of the 40th Annual Meeting on Association for Computational Linguistics (pp. 152-159).

Qayyum Ul Zia and Altaf Wasif. 2012. *Paraphrase Identification using Semantic Heuristic Features.* Research Journal of Applied Sciences, Engineering and Technology 4(22): 4894-4904.

Richard Socher., Eric H. Huang., Jeffrey Pennin., Christopher D. Manning and Andrew Y. Ng. 2011. *Dynamic pooling and unfolding recursive autoencoders for paraphrase detection.* Advances in Neural Information Processing Systems (pp. 801-809).

Ronan Collobert and Jason Weston. 2008. *A unified architecture for natural language processing: Deep neural networks with multitask learning.* Proceedings of the 25th international conference on Machine learning (pp. 160-167).

Samuel Fernando and Mark Stevenson. 2008. *A semantic similarity approach to paraphrase detection*. Computational Linguistics UK (CLUK 2008) 11th Annual Research Colloquium (pp. 45-52).

Sasa Petrovic., Miles Osborne and Victor Lavrenko. 2012. *Using paraphrases for improving first story detection in news and Twitter.* Proceedings of the 2012 Conference of the North American Chapter of the Association for Computational Linguistics: Human Language Technologies (pp. 338-346).

Wang Ling., Chris Dyer., Alan W. Black and Isabel Trancoso. 2013. *Paraphrasing 4 Microblog Normalization.* In EMNLP (pp. 73-84).

Wei Wu., Yun-Cheng Ju., Xiao Li and Ye-Yi Wang. 2010. *Paraphrase detection on SMS messages in automobiles.* In Acoustics Speech and Signal Processing (ICASSP), 2010 IEEE International Conference on (pp. 5326-5329).

Wei Xu., Alan Ritter., Bill Dolan., Ralph Grishman and Colin Cherry. 2012. *Paraphrasing for Style.* Proceedings of COLING 2012:Technical paper, pages 2899-2914, Coling 2012, Mumbai, December 2012.

Wei Xu., Alan Ritter and Ralph Grishmann. 2013. *Gathering and generating paraphrases from twitter with application to normalization.* Proceedings of the Sixth Workshop on Building and Using Comparable Corpora (pp. 121-128).

Wei Xu., Ralph Grishman., Adam Meyers and Alan Ritter. 2013. *A Preliminary Study of Tweet Summarization using Information Extraction.*

Wei Xu 2014. *Data-driven approaches for paraphrasing across language variations (Doctoral dissertation, New York University).*

Wei Xu., Alan Ritter., Chris Callison Burch., William B. Dolan and Yangfeng Ji. 2014. *Extracting Lexically Divergent Paraphrases from Twitter.* Transactions Of The Association For Computational Linguistics, 2, 435-448.

Wei Xu., Chris Callison Burch and William B. Dolan. 2015. *Paraphrase and Semantic Similarity in Twitter (PIT2015).* International Workshop on Semantic Evaluation (SemEval 2015).