Who's to say what's funny? A computer using Language Models and Deep Learning, That's Who!

Anonymous ACL submission

Abstract

Humor is a defining characteristic of human beings. Our goal is to develop methods that automatically detect humorous statements and rank them on a continuous scale. In this paper we report on results using a Language Model approach, and outline our plans for using methods from Deep Learning.

1 Introduction

Computational humor is an emerging area of research that ties together ideas from psychology, linguistics, and cognitive science. *Humor generation* is the problem of automatically creating humorous statements (e.g., (Stock and Strapparava, 2003), (Özbal and Strapparava, 2012)). *Humor detection* seeks to identify humor in text, and is sometimes cast as a binary classification problem that decides if some input is humorous or not (e.g., (Mihalcea and Strapparava, 2006), (Zhang and Liu, 2014), (Shahaf et al., 2015), (Miller and Gurevych, 2015)). However, our focus is on the continuous and subjective aspects of humor.

We learn a particular sense of humor from a data set of tweets which are geared towards a certain style of humor (Potash et al., 2016). This data consists of humorous tweets which have been submitted in response to hashtag prompts provided during the Comedy Central TV show *@midnight with Chris Hardwick*. Since not all jokes are equally funny, we use Language Models and methods from Deep Learning to allow potentially humorous statements to be ranked relative to each other.

2 Language Models

We used traditional Ngram language models as our first approach for two reasons : First, Ngram language models can learn a certain style of humor by using examples of that as the training data for the model. Second, they assign a probability to each input they are given, making it possible to rank statements relative to each other. Thus, Ngram language models make relative rankings of humorous statements based on a particular style of humor, thereby accounting for the continuous and subjective nature of humor.

We began this research by participating in SemEval-2017 Task 6 #HashtagWars: Learning a Sense of Humor (Potash et al., 2017). This included two subtasks : Pairwise Comparison (Subtask A) and Semi-ranking (Subtask B). Pairwise comparison asks a system to choose the funnier of two tweets. Semi-ranking requires that each of the tweets associated with a particular hashtag be assigned to one of the following categories : top most funny tweet, next nine most funny tweets, and all remaining tweets.

Our system estimated tweet probabilities using Ngram language models. We created models from two different corpora - a collection of funny tweets from the @midnight program, and a corpus of news data that is freely available for research[1]. We scored tweets by assigning them a probability based on each model. Tweets that have a higher probability according to the funny tweet model are considered funnier since they are more like the humorous training data. However, tweets that have a lower probability according to the news language model are viewed as funnier since they are least like the (unfunny) news corpus. We took a standard approach to language modeling and used bi-

[1] http://www.statmt.org/wmt11/featured-translation-task.html

grams and trigrams as features in our models. We used KenLM (Heafield et al., 2013) with modified Kneser-Ney smoothing and a back-off technique as our language modeling tool.

Table 1 shows our results for both data sets when trained on bigrams and trigrams. The accuracy and distance measures are defined by the task organizers (Potash et al., 2017). We seek high accuracy in picking the funnier tweet (Subtask A) and low distance (from the gold standard) in organizing the tweets into categories (Subtask B).

Data	Ngram	Accuracy (A)	Distance (B)
tweets	trigram	0.397	0.967
tweets	bigram	0.406	0.944
news	trigram	0.627	0.872
news	bigram	0.624	0.853

Table 1: Experimental results

These results show that models trained on the news data have a significant advantage over the tweets model, and that bigram models performed slightly better than trigrams. We submitted trigram models trained on news and tweets to the official evaluation of SemEval-2017 Task 6. The trigram language models trained on the news data placed fourth in Subtask A and first in Subtask B.

We believe that the significant advantage of the news data over the tweet data is caused by the much larger quantity of news data available. The tweet data only consists of approximately 21,000 tweets, whereas the news data totals approximately 6.2 GB of text. In the future we intend to collect more tweet data, especially those participating in the ongoing #HashtagWars staged nightly by @midnight. We also plan to experiment with equal amounts of tweet data and news data, to see if one has an inherent advantage over the other.

Our language models performed better in the pairwise comparison, but it is clear that more investigation is needed to improve the semi-ranking results. We believe that Deep Learning may overcome some of the limits of Ngram language models, and so will explore those next.

3 Deep Learning

One limitation of our language model approach is the large number of out of vocabulary words we encounter. This problem can not be solved by increasing the quantity of training data because humor relies on creative use of language. For example, jokes often include puns based on invented words, e.g., a singing cat makes beautiful *me*owsic. (Potash et al., 2016) suggests that character–based Convolutional Neural Networks (CNNs) are an effective solution for these situations since they are not dependent on observing tokens in training data. Previous work has also shown the CNNs are effective tools for language modeling, even in the presence of complex morphology (Kim et al., 2015). Other recent work has shown that Recurrent Neural Networks (RNNs), in particular Long Short–Term Memory networks (LSTMs), are effective in a wide range of language modeling tasks (e.g., (Sundermeyer et al., 2012),(Sundermeyer et al., 2015)). This seems to be due to their ability to capture long distance dependencies, which is something that Ngram language models can not do.

(Potash et al., 2016) finds that external knowledge is necessary to detect humor in tweet based data. This might include information about book and movie titles, song lyrics, biographies of celebrities etc. and is necessary given the reliance on current events and popular culture in making certain kinds of jokes.

We believe that Deep Learning techniques potentially offer improved handling of unknown words, long distance dependencies in text, and non–linear relationships among words and concepts. Moving forward we intend to explore a variety of these ideas and describe those briefly below.

4 Future Work

Our current language model approach is effective but does not account for out of vocabulary words nor long distance dependencies. CNNs in combination with LSTMs seem to be a particularly promising way to overcome these limitations (e.g., (Bertero and Fung, 2016)) which we will explore and compare to our existing results.

After evaluating CNNs and LSTMs we will explore how to include domain knowledge in these models. One possibility is to create word embeddings from domain specific materials and provide those to the CNNs along with more general text. Another is to investigate the use of Tree–Structured LSTMs (Tai et al., 2015). These have the potential advantage of preserving non-linear structure in text, which may be helpful in recognizing some of the unusual variations of words and concepts that are characteristic of humor.

References

Dario Bertero and Pascale Fung. 2016. A long short-term memory framework for predicting humor in dialogues. In *Proceedings of the 2016 Conference of the North American Chapter of the Association for Computational Linguistics: Human Language Technologies*. Association for Computational Linguistics, San Diego, California, pages 130–135.

Kenneth Heafield, Ivan Pouzyrevsky, Jonathan H. Clark, and Philipp Koehn. 2013. Scalable modified Kneser-Ney language model estimation. In *Proceedings of the 51st Annual Meeting of the Association for Computational Linguistics*. Sofia, Bulgaria, pages 690–696.

Yoon Kim, Yacine Jernite, David Sontag, and Alexander M Rush. 2015. Character-aware neural language models. *arXiv preprint arXiv:1508.06615* .

Rada Mihalcea and Carlo Strapparava. 2006. Learning to laugh (automatically): Computational models for humor recognition. *Computational Intelligence* 22(2):126–142.

Tristan Miller and Iryna Gurevych. 2015. Automatic disambiguation of english puns. In *Proceedings of the 53rd Annual Meeting of the Association for Computational Linguistics and the 7th International Joint Conference on Natural Language Processing (Volume 1: Long Papers)*. Association for Computational Linguistics, Beijing, China, pages 719–729.

Gözde Özbal and Carlo Strapparava. 2012. A computational approach to the automation of creative naming. In *Proceedings of the 50th Annual Meeting of the Association for Computational Linguistics: Long Papers-Volume 1*. Association for Computational Linguistics, pages 703–711.

Peter Potash, Alexey Romanov, and Anna Rumshisky. 2016. #HashtagWars: Learning a sense of humor. *arXiv preprint arXiv:1612.03216* .

Peter Potash, Alexey Romanov, and Anna Rumshisky. 2017. SemEval-2017 Task 6: #HashtagWars: learning a sense of humor. In *Proceedings of the 11th International Workshop on Semantic Evaluation (SemEval-2017)*. Vancouver, BC.

Dafna Shahaf, Eric Horvitz, and Robert Mankoff. 2015. Inside jokes: Identifying humorous cartoon captions. In *Proceedings of the 21th ACM SIGKDD International Conference on Knowledge Discovery and Data Mining*. ACM, New York, NY, USA, KDD '15, pages 1065–1074.

Oliviero Stock and Carlo Strapparava. 2003. Getting serious about the development of computational humor. In *Proceedings of the Eighteenth International Joint Conference on Artificial Intelligence*. Acapulco, pages 59–64.

Martin Sundermeyer, Hermann Ney, and Ralf Schlüter. 2015. From feedforward to recurrent lstm neural networks for language modeling. *IEEE/ACM Transactions on Audio, Speech and Language Processing (TASLP)* 23(3):517–529.

Martin Sundermeyer, Ralf Schlüter, and Hermann Ney. 2012. Lstm neural networks for language modeling. In *Interspeech*. pages 194–197.

Kai Sheng Tai, Richard Socher, and Christopher D. Manning. 2015. Improved semantic representations from tree-structured long short-term memory networks. In *Proceedings of the 53rd Annual Meeting of the Association for Computational Linguistics and the 7th International Joint Conference on Natural Language Processing (Volume 1: Long Papers)*. Association for Computational Linguistics, Beijing, China, pages 1556–1566.

Renxian Zhang and Naishi Liu. 2014. Recognizing humor on twitter. In *Proceedings of the 23rd ACM International Conference on Conference on Information and Knowledge Management*. ACM, New York, NY, USA, CIKM '14, pages 889–898.