Pretraining Sentiment Classifiers with Unlabeled Dialog Data

Toru Shimizu[1], **Hayato Kobayashi**[1,2], **and Nobuyuki Shimizu**[1]

[1]Yahoo Japan Corporation
[2]Riken AIP
{toshimiz,hakobaya,nobushim}@yahoo-corp.jp

Abstract

The huge cost of creating labeled training data is a common problem for supervised learning tasks such as sentiment classification. Recent studies showed that pretraining with unlabeled data via a language model can improve the performance of classification models. In this paper, we take the concept a step further by using a conditional language model, instead of a language model. Specifically, we address a sentiment classification task for a tweet analysis service as a case study and propose a pretraining strategy with unlabeled dialog data (tweet-reply pairs) via an encoder-decoder model. Experimental results show that our strategy can improve the performance of sentiment classifiers and outperform several state-of-the-art strategies including language model pretraining.

1 Introduction

Sentiment classification is a task to predict a sentiment label, such as positive/negative, for a given text and has been applied to many domains such as movie/product reviews, customer surveys, news comments, and social media. A common problem of this task is the lack of labeled training data due to costly annotation work, especially for social media without explicit sentiment feedback such as review scores.

To overcome this problem, Dai and Le (2015) recently proposed a semi-supervised sequence learning framework, where a sentiment classifier based on recurrent neural networks (RNNs) is trained with labeled data after initializing it with the parameters of an RNN-based language model pretrained with a large amount of unlabeled data. The concept of their framework is simple but effective, and their work yielded many related studies of semi-supervised training based on sequence modeling, as described in Section 4.

In this paper, we take their concept a step further by using a conditional language model with unlabeled dialog data (i.e., tweet-reply pairs) instead of a language model with unpaired data[1]. An important observation of the dialog data that underpins our strategy is that the sentiment or mood in a message often affects messages in reply to it. People tend to write angry responses to angry messages, empathetic replies to sad remarks, or congratulatory phrases to good news.

Our contributions are listed as follows.

- We propose a pretraining strategy with unlabeled dialog data (tweet-reply pairs) via an encoder-decoder model for sentiment classifiers (Section 2). To the best of our knowledge, our proposal is the first such proposal, as clarified in Section 4.
- We report on a case study based on a costly labeled sentiment dataset of 99.5K items and a large-scale unlabeled dialog dataset of 22.3M, which were provided from a tweet analysis service (Section 3.1).
- Experimental results of sentiment classification show that our method outperforms the current semi-supervised methods based on a language model, autoencoder, and distant supervision, as well as linear classifiers (Section 3.4).

2 Proposed Method

Our pretraining strategy simply consists of the following two steps:

[1]We use the term "conditional language model" in a narrow sense only for a model trained with explicit source-target pairs, although both RNN-based language and autoencoder models can generate a text from a real-valued context vector.

Proceedings of the 56th Annual Meeting of the Association for Computational Linguistics (Short Papers), pages 764–770
Melbourne, Australia, July 15 - 20, 2018.

1. Training a dialog (encoder-decoder) model using unlabeled dialog data (tweet-reply pairs) as pretraining.

2. Training a sentiment classifier (encoder-labeler) model using labeled sentiment data (tweet-label pairs) after initializing its encoder part with the encoder parameters of the encoder-decoder model.

The encoder-decoder model is a conditional language model that predicts a correct output sequence from an input sequence (Sutskever et al., 2014). This model consists of two RNNs: an encoder and decoder. The encoder extracts a context of the input sequence as a real-valued vector, and the decoder predicts the output sequences from the context individually.

Our classifier forms an encoder-labeler structure, which consists of the above encoder and a labeler that predicts a sentiment label from the context. Note that the encoder of the classifier is fine-tuned with labeled data, as in (Dai and Le, 2015). The main difference between their approach and ours is that we examine paired (dialog) data for pretraining, while they only showed the usefulness of pretraining with unpaired data.

3 Experiments

3.1 Datasets

We used two datasets, a dialog dataset for pretraining the encoder-decoder model and a sentiment dataset for training (fine-tuning) the sentiment classifier, as shown in Table 1. Those datasets were provided by Yahoo! JAPAN, which is the largest portal site in Japan.

The dialog dataset contains about 22.3 million tweet-reply pairs extracted from Twitter Firehose data. In its preprocessing, we filtered out spam and bot posts by using user-level signals such as the follower count, the friend count, the favorite count, and whether a profile image is set or not. Also, we replaced all the URLs in the text with "[u]" and all the user mentions with "[m]", considering them as noise. The rest of the text was used as it was. On average, source and target (or reply) tweets after preprocessing were 31.5 and 27.8 characters long, respectively. While redistribution of tweets is prohibited, we are planning to publicize tweet IDs of this dataset for reproducibility.[2]

	Train	Valid	Test
Dialog	22,300,000	10,000	50,000
Sentiment	80,591	4,000	15,000

Table 1: Details of dialog and sentiment datasets

The sentiment dataset includes about 100K tweets with manually annotated three-class sentiment labels: `positive`, `negative`, and `neutral`. The breakdown of `positive`, `negative`, and `neutral` in the training set was 15.0, 18.6, and 66.4%, respectively. Note that the tweets were sampled separately from those of the dialog dataset. The procedure for text preprocessing was the same with that of the dialog dataset. The average length of the tweets after preprocessing was 17 characters. Each tweet was judged by a majority vote of three experienced editors in the company providing the sentiment-analysis service. The inter-annotator agreement ratio assessed with Fleiss' κ was 0.495. The overall annotation work took roughly 300 person-days. This means that the cost is at least 24K dollars, 8 hours $\times$ 300 days $\times$ legal minimum wage in Japan 10 dollars/hour. Considering that the in-house annotators are well-educated, skilled proper employees, the actual cost would be much higher than this rough estimate and much more costly than collecting unlabeled dialog data. In addition, the annotators had gone through a few days of training to become able to appropriately judge the sentiment before they got down to actual annotation work, but the number, 300 person-days, does not include the time for this training.

3.2 Model and Training

The settings of the dialog (encoder-decoder) model are as follows. In both the encoder and decoder, the size of the word-embedding layer is 256 and that of the LSTM-RNN hidden layer is 1024. The size of the output layer is 4000, which is the same as the (character-based) vocabulary size.[3]. The encoder and decoder share these hyperparameters as well as the parameters themselves (that is, with regard to the embedding layer and

[2]The tweet IDs will be provided from https://research-lab.yahoo.co.jp/en/software/ .

[3]We used a character-based model since it performed better than word-based models in our preliminary experiments. Existing morphological analyzers needed for word-based models have usually been trained by formal text such as that of newspapers and seem not suitable to highly colloquial text seen in tweets, which often includes emoticons and emoji.

recurrent layer). The total number of parameters is 8.9 million.

The settings of the sentiment classifier (encoder-labeler) model are as follows. The encoder part has the same structure and hyper-parameters as that of the dialog model, making them compatible for transferring learned parameters. We reused the dialog model's dictionaries in the classifier model so that the two models could process tweet texts consistently. The labeler consists of a fully connected layer and soft max nonlinearity.

The models were trained with ADADELTA (Zeiler, 2012) with a mini-batch size of 64. The dialog model was trained in five epochs, and the classifier model was tuned with the early-stopping strategy, which stops training when the validation accuracy drops. For ADADELTA's parameters, we fixed the learning rate to 1.0, decay rate ρ to 0.95, and smoothing constant ϵ to 10^{-6} for all training sessions. We evaluated validation costs ten times per epoch and selected the model with the lowest validation cost. The training took 15.9 days on 1 GPU with 7 TFLOPS computational power.

3.3 Compared Models

We compared the following eight models: non-pretrained (`Default`), proposed dialog pretraining (`Dial`), current pretraining with unpaired data (`Lang`, `SeqAE`) and pseudo labeled data (`Emo2M`, `Emo6M`), and classical linear learners (`LogReg`, `LinSVM`). The details of these models are given below.

- `Default`: Trained without pretraining by executing only Step 2 in Section 2.

- `Dial`: Pretrained with the dialog model described in Section 2.

- `Lang`, `SeqAE`: Pretrained with the language model and autoencoder model proposed in (Dai and Le, 2015). The language model is the decoder part of the encoder-decoder model using a zero vector as the initial hidden layer value, and the autoencoder model is the same structure of the encoder-decoder model, where input and output are the same. To make the comparison as fair as possible, we used the reply-side of the dialog dataset for pretraining `Lang` and `SeqAE` so that the same supervision information on the basis of the same tweet-reply pairs would be applied to `Lang`, `SeqAE`, and `Dial`. The number of their pretraining epochs was also equal to that of `Dial`.

- `Emo2M`, `Emo6M`: Pretrained with pseudo labeled data (2M, 6M) based on manually collected emoticons, which consist of 120 positive emoticons and 116 negative ones. This technique is also known as distant-supervision. These pseudo labels were annotated by extracting tweets including one of those emoticons from our dialog data and another 92M tweets. Pretraining was conducted via a two-class sentiment classifier, which is a similar model to `Default`, since uncertain tweets without emoticons are not always `neutral`. We confirmed that this two-class classifier can reach more than 90% test accuracy on the emoticon-based test dataset. After pretraining, the parameters of the encoder part were transfered to the final classifier model.

- `LogReg`, `LinSVM`: Logistic regression and linear support vector machine (SVM) models of LIBLINEAR (Fan et al., 2008) with bag-of-words features, which consist of 50K unigrams (w/o stopwords), 50K bigrams, and 233 emoticons. These features are based on a state-of-the-art system (Mohammad et al., 2013) that performed best in the SEMEVAL competition (Nakov et al., 2013) and was actually used in the tweet analysis service of the data-providing company. The best parameters were found through a grid-search on the validation set.

3.4 Results

Table 2 shows the macro-average F-measure results of the compared models in Section 3.3 on the sentiment classification task when varying data size (5K to 80K). Each value is the average of five trials with different random seeds for each setting, and a value of a trial is the macro-average of F-measure values of three sentiment classes. The first row (`Default`) shows the default sentiment classifier model without pretraining. The second row block (`Dial` to `Emo6M`) shows the results of the same training as `Default` after pretraining via different models, while the third block shows those of linear classifiers (non-RNN models). The supplemental materials also include the results measured by accuracy.

	5K	10K	20K	40K	80K
Default	0.517	0.590	0.623	0.653	0.673
Dial	**0.665**†	**0.685**†	**0.702**†	**0.717**†	**0.738**†
Lang	0.653	0.674	0.692	0.707	0.726
SeqAE	0.568	0.598	0.626	0.649	0.677
Emo2M	0.482	0.532	0.579	0.626	0.664
Emo6M	0.484	0.517	0.565	0.613	0.650
LogReg	0.577	0.609	0.631	0.648	0.675
LinSVM	0.582	0.610	0.627	0.637	0.648

Table 2: Macro-average F-measure of sentiment classification of each model versus labeled data size. Dial is our proposed method, and † in its row indicates statistically significant difference from the corresponding value of Lang ($p < 0.05$).

Comparing Dial with the other models, we can see that our pretraining strategy with dialog data consistently outperformed all the other models: state-of-the-art pretraining strategies with unpaired unlabeled data (Lang, SeqAE) and pseudo labeled data (Emo2M, Emo6M), as well as linear learners (LogReg, LinSVM). This indicates that unlabeled dialog data (tweet-reply pairs) have useful information for sentiment classifiers, as expected in Section 1. In fact, we observed that the pretrained encoder-decoder model seems to generate an appropriate reply, on which the sentiment on the input tweet is well reflected. For example, the reply ":(" was generated for the input tweet "I'm sorry to hear that" (see supplementary material for more examples).

Lang also outperformed well but did not overtake Dial. The differences between Dial and Lang are statistically significant[4] for all five training dataset sizes. Interestingly, SeqAE was not so effective like Dial, despite their model structures are basically the same. This implies that it is practically important to find appropriate data for pretraining, such as dialog data for sentiment classification.

As for the results of distant supervision with emoticons, both Emo2M and Emo6M performed worse than Default, and increasing the dataset size did not change the situation. The reason why these models did not perform as well as other pretraining-based models is considered to be noisy labels, especially in negative ones. We illustrate two instances in the Emo2M training data that include an emoticon that is usually negative emoticon but can be considered positive:

- 美人すぎるよ可愛い(;;), "She is so beautiful, cute (crying emoticon)"
- うらやましいです。おめでとうございますorz, "I envy you. Congratulations (bow-the-knee emoticon)"

Comparing Default with LogReg and LinSVM, we can see that the linear models performed better than the default RNN model without pretraining, when the labeled data size is less than or equal to 20K. However, looking at the results of Dial, our method improved Default even for these cases (5K to 20K), and Dial clearly outperformed the linear models. This means that pretraining is useful especially on the situation where the labeled data size is limited.

4 Related Work

After Dai and Le (2015) proposed the framework of semi-supervised sequence learning, there have been several attempts to extend sequence learning models for different tasks to semi-supervised settings. Cheng et al. (2016) and Ramachandran et al. (2017) studied semi-supervised training of machine translation models via an autoencoder model and language model, respectively. They also used paired data (parallel corpora), but unsupervised training was conducted with reasonable monolingual corpora to compensate for costly parallel corpora, which is opposite to our setting. Zhou et al. (2016a,b) proposed to use parallel corpora for adapting the sentiment resources in a resource-rich language to a resource-poor language. Their purpose was completely different from ours, since making parallel corpora is also costly. The other studies include semi-supervised extensions for predicting the property values of Wikipedia (Hewlett et al., 2017), detecting medical conditions from heart rate data (Ballinger et al., 2018), and morphological reinflection of inflected words (e.g., "playing" to "played"). They did not use paired-text data to leverage their tasks.

Our method can be regarded as a general version of distant supervision since we assume that a reply includes the label information of the corresponding tweet. There have been many studies about distant supervision for sentiment analysis (Read, 2005; Go et al., 2009; Davidov et al., 2010; Purver and Battersby, 2012; Mohammad et al., 2013; Tang et al., 2014; dos Santos and Gatti,

[4]Under the significance level of 0.05 with two-tailed t-test assuming unequal variances.

2014; Severyn and Moschitti, 2015; Deriu et al., 2016; Müller et al., 2017), but they basically focused on how to use emoticons and hashtags to leverage performance. One exception is the study by (Pool and Nissim, 2016), in which Facebook reactions were used for distant supervision. Their approach is similar to ours using tweet-reply pairs, but our method is more general since they only used six reply categories (i.e., like, love, haha, wow, sad, and angry), not text replies.

There have been a few studies on sentiment classification in dialogue data (Bertero and Fung, 2016; Bertero et al., 2016). These studies involved sentiment classification based on dialog contexts, which means that they used labeled dialog data, while we used unlabeled dialog data. For tweet data, several studies used reply-features for sentiment classification of tweets (Barbosa and Feng, 2010; Jiang et al., 2011; Vanzo et al., 2014; Bamman and Smith, 2015; Ren et al., 2016; Castellucci et al., 2016). However, they used replies as labeled data for sentiment classification, not unlabeled data for pretraining.

5 Conclusion

We proposed a pretraining strategy with dialog data for sentiment classifiers. The experimental results showed that our strategy clearly outperformed the existing pretraining with unpaired unlabeled data via language modeling and pseudo labeled data via distant supervision, as well as linear classifiers. In the future, we will investigate whether or not we can use other paired data for pretraining of classification tasks. For example, we expect that news article-comment pairs are useful for predicting fake news detection and that question-answer pairs of Q&A sites are useful for recommending questions for answering.

References

Brandon Ballinger, Johnson Hsieh, Avesh Singh, Nimit Sohoni, Jack Wang, Geoffrey H. Tison, Gregory M. Marcus, Jose M. Sanchez, Carol Maguire, Jeffrey E. Olgin, and Mark J. Pletcher. 2018. DeepHeart: Semi-Supervised Sequence Learning for Cardiovascular Risk Prediction. In *Proceedings of the Thirty-Second AAAI Conference on Artificial Intelligence (AAAI 2018)*. To appear. https://arxiv.org/abs/1802.02511.

David Bamman and Noah A. Smith. 2015. Contextualized Sarcasm Detection on Twitter. In *Proceedings of the Ninth International Conference on Web and Social Media (ICWSM 2015)*. pages 574–577. http://www.aaai.org/ocs/index.php/ICWSM/ICWSM15/paper/view/10538.

Luciano Barbosa and Junlan Feng. 2010. Robust Sentiment Detection on Twitter from Biased and Noisy Data. In *Proceedings of the 21st International Conference on Computational Linguistics (COLING 2010)*. Coling 2010 Organizing Committee, pages 36–44. http://www.aclweb.org/anthology/C10-2005.

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 (NAACL-HLT 2016)*. Association for Computational Linguistics, pages 130–135. http://www.aclweb.org/anthology/N16-1016.

Dario Bertero, Farhad Bin Siddique, Chien-Sheng Wu, Yan Wan, Ricky Ho Yin Chan, and Pascale Fung. 2016. Real-Time Speech Emotion and Sentiment Recognition for Interactive Dialogue Systems. In *Proceedings of the 2016 Conference on Empirical Methods in Natural Language Processing (EMNLP 2016)*. Association for Computational Linguistics, pages 1042–1047. https://aclweb.org/anthology/D16-1110.

Giuseppe Castellucci, Danilo Croce, and Roberto Basili. 2016. Context-aware Convolutional Neural Networks for Twitter Sentiment Analysis in Italian. In *Proceedings of Third Italian Conference on Computational Linguistics (CLiC-it 2016) & Fifth Evaluation Campaign of Natural Language Processing and Speech Tools for Italian. Final Workshop (EVALITA 2016)*. http://ceur-ws.org/Vol-1749/paper_029.pdf.

Yong Cheng, Wei Xu, Zhongjun He, Wei He, Hua Wu, Maosong Sun, and Yang Liu. 2016. Semi-Supervised Learning for Neural Machine Translation. In *Proceedings of the 54th Annual Meeting of the Association for Computational Linguistics (ACL 2016)*. Association for Computational Linguistics, pages 1965–1974. http://www.aclweb.org/anthology/P16-1185.

Andrew M Dai and Quoc V Le. 2015. Semi-supervised Sequence Learning. In *Advances in Neural Information Processing Systems 28 (NIPS 2015)*, Curran Associates, Inc., pages 3079–3087. http://papers.nips.cc/paper/5949-semi-supervised-sequence-learning.pdf.

Dmitry Davidov, Oren Tsur, and Ari Rappoport. 2010. Enhanced Sentiment Learning Using Twitter Hashtags and Smileys. In *Proceedings of the 21st International Conference on Computational Linguistics (COLING 2010)*. Coling 2010 Organizing Committee, pages 241–249. http://www.aclweb.org/anthology/C10-2028.

Jan Deriu, Maurice Gonzenbach, Fatih Uzdilli, Aurelien Lucchi, Valeria De Luca, and Martin Jaggi. 2016. SwissCheese at SemEval-2016 Task 4: Sentiment Classification Using an Ensemble of Convolutional Neural Networks with Distant Supervision. In *Proceedings of the 10th International Workshop on Semantic Evaluation (SemEval 2016)*. Association for Computational Linguistics, pages 1124–1128. http://www.aclweb.org/anthology/S16-1173.

Cicero dos Santos and Maira Gatti. 2014. Deep Convolutional Neural Networks for Sentiment Analysis of Short Texts. In *Proceedings of the 25th International Conference on Computational Linguistics (COLING 2014)*. Dublin City University and Association for Computational Linguistics, pages 69–78. http://www.aclweb.org/anthology/C14-1008.

Rong-En Fan, Kai-Wei Chang, Cho-Jui Hsieh, Xiang-Rui Wang, and Chih-Jen Lin. 2008. LIBLINEAR: A Library for Large Linear Classification. *Journal of Machine Learning Research* 9:1871–1874.

Alec Go, Richa Bhayani, and Lei Huang. 2009. Twitter Sentiment Classification using Distant Supervision. Technical report, Stanford Digital Library Technologies Project. https://cs.stanford.edu/people/alecmgo/papers/TwitterDistantSupervision09.pdf.

Daniel Hewlett, Llion Jones, Alexandre Lacoste, and izzeddin gur. 2017. Accurate Supervised and Semi-Supervised Machine Reading for Long Documents. In *Proceedings of the 2017 Conference on Empirical Methods in Natural Language Processing (EMNLP 2017)*. Association for Computational Linguistics, pages 2011–2020. https://www.aclweb.org/anthology/D17-1214.

Long Jiang, Mo Yu, Ming Zhou, Xiaohua Liu, and Tiejun Zhao. 2011. Target-dependent Twitter Sentiment Classification. In *Proceedings of the 49th Annual Meeting of the Association for Computational Linguistics: Human Language Technologies (ACL-HLT 2011)*. Association for Computational Linguistics, pages 151–160. http://www.aclweb.org/anthology/P11-1016.

Saif Mohammad, Svetlana Kiritchenko, and Xiaodan Zhu. 2013. NRC-Canada: Building the State-of-the-Art in Sentiment Analysis of Tweets. In *Proceedings of the Seventh International Workshop on Semantic Evaluation (SemEval 2013)*. Association for Computational Linguistics, pages 321–327. http://www.aclweb.org/anthology/S13-2053.

Simon Müller, Tobias Huonder, Jan Deriu, and Mark Cieliebak. 2017. TopicThunder at SemEval-2017 Task 4: Sentiment Classification Using a Convolutional Neural Network with Distant Supervision. In *Proceedings of the 11th International Workshop on Semantic Evaluation (SemEval 2017)*. Association for Computational Linguistics, pages 766–770. http://www.aclweb.org/anthology/S17-2129.

Preslav Nakov, Sara Rosenthal, Zornitsa Kozareva, Veselin Stoyanov, Alan Ritter, and Theresa Wilson. 2013. SemEval-2013 Task 2: Sentiment Analysis in Twitter. In *Proceedings of the Seventh International Workshop on Semantic Evaluation (SemEval 2013)*. Association for Computational Linguistics, pages 312–320. http://www.aclweb.org/anthology/S13-2052.

Chris Pool and Malvina Nissim. 2016. Distant supervision for emotion detection using facebook reactions. In *Proceedings of the Workshop on Computational Modeling of People's Opinions, Personality, and Emotions in Social Media (PEOPLES)*. The COLING 2016 Organizing Committee, pages 30–39. http://aclweb.org/anthology/W16-4304.

Matthew Purver and Stuart Battersby. 2012. Experimenting with Distant Supervision for Emotion Classification. In *Proceedings of the 13th Conference of the European Chapter of the Association for Computational Linguistics (ACL 2012)*. Association for Computational Linguistics, pages 482–491. http://www.aclweb.org/anthology/E12-1049.

Prajit Ramachandran, Peter Liu, and Quoc Le. 2017. Unsupervised Pretraining for Sequence to Sequence Learning. In *Proceedings of the 2017 Conference on Empirical Methods in Natural Language Processing (EMNLP 2017)*. Association for Computational Linguistics, pages 383–391. https://www.aclweb.org/anthology/D17-1039.

Jonathon Read. 2005. Using Emoticons to Reduce Dependency in Machine Learning Techniques for Sentiment Classification. In *Proceedings of the ACL Student Research Workshop*. Association for Computational Linguistics, pages 43–48. http://www.aclweb.org/anthology/P/P05/P05-2008.

Yafeng Ren, Yue Zhang, Meishan Zhang, and Donghong Ji. 2016. Context-sensitive Twitter Sentiment Classification Using Neural Network. In *Proceedings of the Thirtieth AAAI Conference on Artificial Intelligence (AAAI 2016)*. AAAI Press, pages 215–221. http://dl.acm.org/citation.cfm?id=3015812.3015844.

Aliaksei Severyn and Alessandro Moschitti. 2015. UNITN: Training Deep Convolutional Neural Network for Twitter Sentiment Classification. In *Proceedings of the 9th International Workshop on Semantic Evaluation (SemEval 2015)*. Association for Computational Linguistics, pages 464–469. http://www.aclweb.org/anthology/S15-2079.

Ilya Sutskever, Oriol Vinyals, and Quoc V Le. 2014. Sequence to sequence learning with neural networks. In *Advances in Neural Information Processing Systems 27 (NIPS 2014)*, Curran Associates, Inc., pages 3104–3112. http://papers.nips.cc/paper/5346-sequence-to-sequence-learning-with-neural-networks.pdf.

Duyu Tang, Furu Wei, Nan Yang, Ming Zhou, Ting Liu, and Bing Qin. 2014. Learning

Sentiment-Specific Word Embedding for Twitter Sentiment Classification. In *Proceedings of the 52nd Annual Meeting of the Association for Computational Linguistics (ACL 2014)*. Association for Computational Linguistics, pages 1555–1565. http://www.aclweb.org/anthology/P14-1146.

Andrea Vanzo, Danilo Croce, and Roberto Basili. 2014. A context-based model for Sentiment Analysis in Twitter. In *Proceedings of the 25th International Conference on Computational Linguistics (COLING 2014)*. Dublin City University and Association for Computational Linguistics, pages 2345–2354. http://www.aclweb.org/anthology/C14-1221.

Matthew D. Zeiler. 2012. ADADELTA: An Adaptive Learning Rate Method. *CoRR* abs/1212.5701.

Xinjie Zhou, Xiaojun Wan, and Jianguo Xiao. 2016a. Attention-based LSTM Network for Cross-Lingual Sentiment Classification. In *Proceedings of the 2016 Conference on Empirical Methods in Natural Language Processing (EMNLP 2016)*. Association for Computational Linguistics, Austin, Texas, pages 247–256. https://aclweb.org/anthology/D16-1024.

Xinjie Zhou, Xiaojun Wan, and Jianguo Xiao. 2016b. Cross-Lingual Sentiment Classification with Bilingual Document Representation Learning. In *Proceedings of the 54th Annual Meeting of the Association for Computational Linguistics (ACL 2016)*. Association for Computational Linguistics, pages 1403–1412. http://www.aclweb.org/anthology/P16-1133.