

# Incorporating Context and Knowledge for Better Sentiment Analysis of Narrative Text

Chenyang Lyu  
School of Computing, Dublin City University  
Dublin, Ireland  
chenyang.lyu2@mail.dcu.ie

Tianbo Ji  
ADAPT Centre, Dublin City University  
Dublin, Ireland  
tianbo.ji2@mail.dcu.ie

Yvette Graham  
ADAPT Centre, Dublin City University  
Dublin, Ireland  
yvette.graham@dcu.ie

## Abstract

Recent years have witnessed a significant increase in the availability of online narrative texts, such as news articles and online stories. Automatic sentiment classification of such narrative texts will enable better organisation, search and retrieval. Little research has been carried out to date that to automatically predict sentiment of narrative text however despite much work being carried out to date on sentiment analysis of social media and user feedback. In this paper, we present an approach to sentiment analysis of narrative text that employs pre-trained language model, an approach already proven effective for a range of other NLP tasks. However, for the purpose of sentiment analysis of narrative text in particular, we introduce two new features: a *contextual* feature and *extra knowledge* feature that prove to aid text understanding for prediction of sentiment. We conduct preliminary experiments on a publicly available dataset of fairy tale texts. Results show an increase of accuracy over vanilla pre-trained language model and baseline models for classification of sentiment for narrative texts.

## 1 Introduction

Sentiment analysis refers to the analysis and prediction of the sentiment and emotion in language. Most research carried out to date in this domain has been based on user-generated content such as social media and user feedback [VC12]. Recently, with the development of online narrative resources such as web-fiction and news, the interest in sentiment analysis of narrative text is slowly expanding [Moh13] in order to make better organisation and retrieval. The sentiment and emotion of narrative text can characterised defined as a general feeling reflected in the text and in this current work we use sentiment to stand for both sentiment and emotion.

Contrary to other types of text, narrative is a perceived sequence of non-randomly connected events [Too12]. For example, *A countryman's son stepped on a snake's tail accidentally. The tail suddenly turned and hit him so that he died. The father was very angry so that he cut off part of the snake's tail.* is a piece of narrative text<sup>1</sup> composed of connected events. Due to its complexity, analysing the sentiment of narrative text requires

---

*Copyright © by the paper's authors. Use permitted under Creative Commons License Attribution 4.0 International (CC BY 4.0).*

In: R. Campos, A. Jorge, A. Jatowt, S. Bhatia (eds.): Proceedings of the Text2Story'20 Workshop, Lisbon, Portugal, 14-April-2020, published at <http://ceur-ws.org>

<sup>1</sup><http://englishjuniorhighschool.blogspot.com/2012/04/countryman-and-snake- countrymans-son.html>

a deep understanding of semantic structure and opinions. Since the events that impact the plot of narrative texts are commonly interrelated, this increases the importance of sufficiently taking context into consideration. Furthermore, there are several linguistic phenomena, such as metaphor in sentence *The lack of having proper standing army is Achilles heel for this small country*, and terminology requiring additional knowledge in order to be correctly interpreted. Incorporating such additional knowledge has the potential to help predict the sentiment of text.

Nowadays, pre-trained language models, such as those in [DCLT19], [PNI<sup>+</sup>18] and [RNSS], have proven effective for a range of tasks including syntactic and semantic tasks, like textual similarity, sentiment classification, textual entailment, etc. These models are usually trained on large scale corpora and have the ability to capture and model semantic dependencies over long text. Such pre-trained models have the potential to detect and classify the sentiment of narrative text.

In this paper, we propose a model that employs a state-of-the-art pre-trained language model to classify the sentiment class of narrative text. This model takes context into consideration as well as additional knowledge as features to predict the sentiment class. Firstly, what we call *context features* are used to generate a context-aware representation of a given sentence, before it is used to determine its sentiment class. We provide a detailed description of the methodology and experiments investigating the accuracy of our approach in Sections 3 and 4, respectively.

## 2 Related Work

Sentiment Analysis is regarded as a text classification or text mining problem, and there are two main approaches to this problem: a rule-based approach and a learning-based approach. At first in sentiment analysis, a rule-based approach was attempted involving pre-defined lexicon with corresponding sentiment polarity to predict the polarity of a given text by capturing word occurrences [HS00, DC07, WWB<sup>+</sup>04]. Besides lexicon methods, Zhang et al [ZZL<sup>+</sup>09] also proposed a document-level sentiment analysis framework, which determines sentiment polarity of sentences based on word dependency before aggregating sentences for document-level sentiment prediction, in which single sentence polarity is computed independent of context.

However, sentiment of text can be expressed in a more subtle manner than is possible to detect with such approaches, since they are difficult to classify only based on indicative lexical features. Machine learning algorithms have the potential to address the shortcomings of rule-based approaches. However, learning-based approaches require a labeled training set to be trained on and then predict new samples. Pang et al [PLV02] proposed three machine learning models, Naïve Bayes, Support Vector Machine and Maximum Entropy to predict sentiment based on bag-of-words feature. [NIK10] proposed a dependency tree-based model to capture interactions between words, while [VCB14] and [KF16] apply a range of kernel methods.

Sentiment analysis with neural networks has a long history. For example, Socher et al [SPW<sup>+</sup>13] proposed Recursive Neural Tensor Networks (RNTNs) to model sentence-level sentiment recursively from its syntactic tree. [dSG14] on the other hand adopted Convolutional Neural Networks (CNNs) to classify sentiment for short text, their model outperforming traditional machine learning models, such as Naïve Bayes and SVM. Some alternate approaches using Recurrent Neural Networks have also been proposed, such as [WHZ<sup>+</sup>16] and [ZWX16], the latter employing attention-based LSTM models.

Recently, large-scale pre-trained models have emerged and have been shown very effective on various tasks [PNI<sup>+</sup>18, RNSS, DCLT19], including reading comprehension and sentiment classification, etc. Pre-trained models are trained on large unlabeled corpora in unsupervised setting, after pre-training they are able to capture rich semantic patterns and syntactic information from text [JSS19]. For instance, the noun modifier (*The former*) in sentence *The 45-year-old former General Electric CO. executive figures it will be easier this time.* can be clearly attended to its noun (*executive*) by self-attention mechanism in BERT [CKLM19]. Pre-trained model is quite suitable for us to model sentiment of text, we therefore employ BERT (Bidirectional Encoder Representation from Transformer) on our task because of its capability to model bidirectional dependencies among text.

## 3 Methodology

Most current state-of-the-art pre-trained language models are based on a neural architecture known as the transformer [VSP<sup>+</sup>17], originally proposed by Vaswani et al. In this study, we use pre-trained model BERT [WDS<sup>+</sup>19]. The BERT [DCLT19] we use has been pre-trained on a large unlabeled corpus containing over 3 billion words. This large volume of data enables it to model complex language dependencies in text and produce

very rich representations of text. The input is represented as  $document = \{s_1, \dots, s_n\}$ ,  $s_i = \{w_{i,1}, \dots, w_{i,m}\}$ , where  $s_i$  is the  $i$ -th sentence in document.

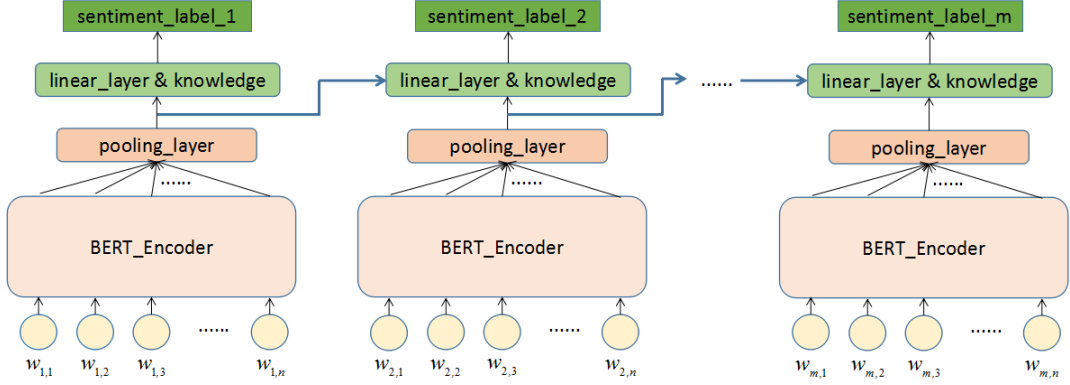


Figure 1: Model Architecture for Sentence-Level Sentiment Classification Incorporating Contextual Feature and Knowledge. The blue arrows stand for contextual vectors from previous sentences.

### 3.1 BERT-Bidirectional Encoder Representation from Transformer

We use BERT as encoder to generate sentiment label for a single sentence, which is based on a transformer block containing one self-attention layer, two layer norms and one feed-forward layer. The transformer we employed consists of 12 stacked transformer blocks, and the output of its last layer is used to get the sentiment label. The overall process of computing sentiment label for a single sentence can be formulated as following:

$$h_l = bert\_encoder(w_1, w_2, \dots, w_k) \quad (1)$$

$$y = softmax(linear\_layer(pooling\_layer(h_l), context)) \quad (2)$$

where  $h_l$  is hidden state vectors from the last layer and it will be fed into pooling layer<sup>2</sup> and linear layer, then passed into softmax layer with  $context$  vector to generate the sentiment label  $y$ , which will be introduced in section 3.2. The overall architecture of our model is shown in Figure 1.

### 3.2 Contextual Feature

Usually, the sentiment label of a single sentence in a document is only conditioned on itself:

$$sentiment\_label = \underset{y}{\operatorname{argmax}} p(y|s_j) \quad (3)$$

which does not take advantage of the context. However, we argue that in narrative, the context of a given sentence plays a crucial role in determining its semantic meaning as well as sentiment label.

We propose to incorporate context features when we compute the sentiment label for a sentence by combining the representation vector of context sentences of a certain sentence, in this way we propose to give sentiment label as following:

$$sentiment\_label = \underset{y}{\operatorname{argmax}} p(y|s_j, context_j) \quad (4)$$

$$context_j = \psi(s_{j-t}, s_{j-t+1}, \dots, s_{j-1}) \quad (5)$$

<sup>2</sup>We use "pooling layer" in accordance with huggingface and this layer simply selects the representation vector of token [CLS], which is a special token added in front of every sentence.

where  $t$  is the number of sentences we incorporate in context,  $\psi$  is the function we use to combine sentence vectors. In our framework, we use the linear combination of context sentences and the target sentence to generate sentiment label. In this work, we use the last sentence of current sentence as it’s context, which has the most significant effect on the sentiment of current sentence.

### 3.3 Extra Knowledge

Besides context feature, we propose to use extra knowledge information to improve understanding of sentiment class of text. In this paper, we incorporate entity information and idiom information into our model. We use idioms dataset SLIDE from [JBBHS18], which includes over 5000 idioms with sentiment annotation. In SLIDE dataset, every idiom has a sentiment distribution, which has three entries. Every entry scales from 0 to 1 and the sum of three entries is 1. For our framework, the extra knowledge is represented by a sentiment-term matrix, every entry of a certain term represents their inclination to corresponding sentiment label and all entries sum to one. The sentiment-term matrix is defined as :

$$M_{i,j} = p(y_j|term_i), \quad \sum_{j=1}^z M_{i,j} = 1 \quad (6)$$

In our model, we incorporate extra knowledge as following:

$$p(y|s_j, context_j) = softmax(normalize(h_j) + \sum_{i=1}^v M_i) \quad (7)$$

$M_i$  is the sentiment distribution vector of the term that is attended in given sentence,  $v$  is the number of all attended terms. We first normalize the hidden state  $h_l$  and then add normalized  $h_l$  and all term vectors  $\{M_i\}_{i=1}^v$  which affect sentiment of text. In our *normalize* function, the hidden state  $h_l$  is normalized and scaled by a factor  $\gamma$ , which makes the ratio of  $l_1$  norms of  $h_l$  and  $M_i$  to  $\gamma$ . In our experiment, we set  $\gamma$  to 10.

## 4 Experiments and Results

### 4.1 Data

For a preliminary validation of our methodology, we evaluate our proposed model with different features on dataset in[Alm08]. This dataset includes 176 fairy tales from three sets of children’s stories: Beatrix Potter(19 stories), H. C. Andersen(77 stories), and the Brothers Grimm(80 stories). There are totally 15,302 sentences annotated with emotional labels in sentence-level. Emotion label set consists of eight labels that are categorized into emotional and non-emotional. The label distribution is in Table.1. The dataset is imbalanced, *NEUTRAL* was most frequent with 66.26%.

Table 1: Label distribution in dataset, where H is *HAPPY*, Su+ is *POSITIVELY SURPRISED*, F is *FEARFUL*, A is *ANGRY*, Sa is *SAD*, D is *DISGUSTED*, Su- is *NEGATIVELY SURPRISED*, N is *NEUTRAL*

Emotional							Non-Emotional
H	Su+	F	A	Sa	D	Su-	N
10.52%	2.15%	4.55%	4.77%	5.43%	3.03%	3.29%	66.26%

### 4.2 Experiment Setup

We firstly split the dataset described in Sec.4.1 into training and test sets. Since in our study we take context into consideration, we select 10% stories from each set of stories as test set. Our test set contains 16 stories, 1,189 sentences in total. The label distribution in training set and test set is in Table.2.

Table 2: Label distribution in training set and test set.

label	H	Su+	F	A	Sa	D	Su-	N
training set	10.58%	2.12%	4.44%	4.32%	5.31%	3.08%	3.19%	66.96%
test set	9.84%	2.52%	5.88%	10.17%	6.90%	2.35%	4.46%	57.86%

The pre-trained model we use is BERT-Base-Cased<sup>3</sup>. The four baselines we select are naïve majority classifier, always predicting the most likely label *NEUTRAL*, and Support Vector Machines (SVM)[CV95] implemented by scikit-learn<sup>4</sup>[PVG<sup>+</sup>11], we use TF-IDF and word2vec embeddings as input to SVM, for word2vec we sum all word embeddings in sentences and then fed it into model. The last baseline is Naïve Bayes. We fine-tune pre-trained model on our training set for 2 epochs, and learning rate of 3e-6, which are recommended settings in original BERT paper[DCLT19].

### 4.3 Classification Results

Table 3 shows our first set of experimental results for classifying sentences to either emotional or non-emotional (i.e., for detecting emotional contents in narratives). In evaluation results, pre-training language model with context feature outperforms all the other models, which proves the effectiveness of incorporating contextual feature. BERT(WF) performs even worse than the naïve classifier, that shows the necessity of fine-tuning. BERT(CF) outperforms vanilla BERT by 2.36% in accuracy and 3.38% in precision, and performs 12.03% better in accuracy and 13.87% in precision over the strongest baselines model Naïve Bayes. We can see from the Table 3. the improvement of incorporating extra knowledge isn't obvious, we argue that may caused by the limited coverage of our knowledge base and the way we incorporate knowledge. Experiments were also run for Bi-LSTMs and CNNs, however the results are not included. Overall, they performed worse than all models listed.

Table 3: Accuracy (%) of emotional versus non-emotional classification and comparison between different features. Our baseline models are naïve majority classifier(NMC), SVM with TF-IDF and Word Embedding, Naïve Bayes. The next 5 models are all BERT-based, in which WF means without fine-tuning, CF means Contextual Feature, EK means Extra Knowledge and CK means Contextual & Knowledge

	Ave. Accuracy (%)	Ave. Precision (%)	Ave. Recall (%)	Ave. F1 (%)
NMC	57.86	28.93	50.00	36.65
SVM+TF-IDF	61.81	63.04	63.15	61.80
SVM+Word2Vec	62.57	64.65	56.99	53.38
Naïve Bayes	62.99	62.30	59.01	57.97
BERT(WF)	57.77	53.35	57.78	43.55
BERT	72.66	72.79	70.33	70.70
BERT(CF)	<b>75.02</b>	<b>76.17</b>	<b>72.37</b>	<b>72.85</b>
BERT(EK)	72.83	73.01	71.12	70.97
BERT(CK)	75.02	76.17	72.37	72.85

Table 4: Evaluation results on Multi-Class classification for emotion labels, metrics are averaged over all classes

	Ave. Accuracy (%)	Ave. Precision (%)	Ave. Recall (%)	Ave. F1 (%)
Naïve Bayes	57.27	36.76	57.27	43.19
SVM+TF-IDF	59.20	51.08	59.21	51.63
SVM+Word2Vec	59.29	57.71	59.29	58.48
BERT	63.07	60.42	63.07	61.71
BERT(CF)	<b>66.28</b>	<b>62.23</b>	<b>66.27</b>	<b>64.19</b>

Furthermore, we conduct experiments on multi-class classification of all labels including *HAPPY*, *POSITIVELY SURPRISED*, *FEARFUL*, *ANGRY*, *SAD*, *DISGUSTED*, *NEGATIVELY SURPRISED*, *NEUTRAL*. Results are shown in Table 4. In the second experiment, we unfortunately did not employ knowledge features because our knowledge base is incompatible with label set. Results in terms of accuracy, precision, recall and F1-score are included. From Table 4, we can see that BERT with contextual feature outperforms the other models in four metrics. The averaged accuracy of BERT(CF) improved by approximately 6.99% over SVM+Word2Vec, by 7.08% over SVM+TF-IDF and by 3.21% over vanilla BERT. Since our dataset is imbalanced, if more annotated data becomes available the performance of our model seems likely to improve.

<sup>3</sup><https://huggingface.co/transformers/>

<sup>4</sup><https://scikit-learn.org/stable/index.html>

## 5 Conclusion and Future Work

In this paper, we apply pre-trained language models to sentence-level sentiment analysis of narrative texts and introduce two new features into model: context feature and extra knowledge feature. We evaluate the performance of a pre-trained model for sentiment analysis of fairy tales with a range of features. Experiments demonstrate that our method improves classification accuracy, showing the importance of incorporating context for sentiment analysis of narrative texts. In this work, we propose a sentence-level model to classify sentiment of narrative texts however how to build a document-level model based on that remains to be further investigated. Our extra knowledge dataset only covers a small portion of text due to limitations with respect to acquisition of the knowledge set with sentiment annotation, hence our incorporation of knowledge did not make a significant improvement. In future work, we aim to build a larger knowledge dataset to improve coverage for narrative text. In addition, we would like to further investigate the alternative ways to inject knowledge into sentiment prediction for narrative text.

## Acknowledgements

This work was supported by Science Foundation Ireland. The authors would like to thank Jennifer Foster and three anonymous reviewers for their helpful comments.

## References

- [Alm08] Cecilia Alm. *Affect in Text and Speech*. PhD thesis, University of Illinois at Urbana-Champaign, 2008.
- [CKLM19] Kevin Clark, Urvashi Khandelwal, Omer Levy, and Christopher D. Manning. What does BERT look at? an analysis of bert’s attention. *CoRR*, abs/1906.04341, 2019.
- [CV95] Corinna Cortes and Vladimir Vapnik. Support-vector networks. *Machine Learning*, 20(3):273–297, Sep 1995.
- [DC07] Sanjiv R. Das and Mike Y. Chen. Yahoo! for amazon: Sentiment extraction from small talk on the web. *Management Science*, 53(9):1375–1388, 2007.
- [DCLT19] Jacob Devlin, Ming-Wei Chang, Kenton Lee, and Kristina Toutanova. BERT: Pre-training of deep bidirectional transformers for language understanding. In *Proceedings of the 2019 Conference of the North American Chapter of the Association for Computational Linguistics: Human Language Technologies, Volume 1 (Long and Short Papers)*, pages 4171–4186, Minneapolis, Minnesota, June 2019. Association for Computational Linguistics.
- [dSG14] Cícero dos Santos and Maíra Gatti. Deep convolutional neural networks for sentiment analysis of short texts. In *Proceedings of COLING 2014, the 25th International Conference on Computational Linguistics: Technical Papers*, pages 69–78, Dublin, Ireland, August 2014. Dublin City University and Association for Computational Linguistics.
- [HS00] Alison Huettner and Pero Subasic. Fuzzy typing for document management. *ACL 2000 Companion Volume: Tutorial Abstracts and Demonstration Notes*, pages 26–27, 2000.
- [JBBHS18] Charles Jochim, Francesca Bonin, Roy Bar-Haim, and Noam Slonim. SLIDE - a sentiment lexicon of common idioms. In *Proceedings of the Eleventh International Conference on Language Resources and Evaluation (LREC 2018)*, Miyazaki, Japan, May 2018. European Language Resources Association (ELRA).
- [JSS19] Ganesh Jawahar, Benoît Sagot, and Djamé Seddah. What does BERT learn about the structure of language? In *Proceedings of the 57th Annual Meeting of the Association for Computational Linguistics*, pages 3651–3657, Florence, Italy, July 2019. Association for Computational Linguistics.
- [KF16] Rasoul Kaljahi and Jennifer Foster. Detecting opinion polarities using kernel methods. In *Proceedings of the Workshop on Computational Modeling of People’s Opinions, Personality, and Emotions in Social Media (PEOPLES)*, pages 60–69, Osaka, Japan, December 2016. The COLING 2016 Organizing Committee.

- [Moh13] Saif Mohammad. From once upon a time to happily ever after: Tracking emotions in novels and fairy tales. *CoRR*, abs/1309.5909, 2013.
- [NIK10] Tetsuji Nakagawa, Kentaro Inui, and Sadao Kurohashi. Dependency tree-based sentiment classification using crfs with hidden variables. In *Human Language Technologies: The 2010 Annual Conference of the North American Chapter of the Association for Computational Linguistics*, pages 786–794. Association for Computational Linguistics, 2010.
- [PLV02] Bo Pang, Lillian Lee, and Shivakumar Vaithyanathan. Thumbs up?: Sentiment classification using machine learning techniques. In *Proceedings of the ACL-02 Conference on Empirical Methods in Natural Language Processing - Volume 10, EMNLP '02*, pages 79–86, Stroudsburg, PA, USA, 2002. Association for Computational Linguistics.
- [PNI<sup>+</sup>18] Matthew E Peters, Mark Neumann, Mohit Iyyer, Matt Gardner, Christopher Clark, Kenton Lee, and Luke Zettlemoyer. Deep contextualized word representations. In *Proceedings of NAACL-HLT*, pages 2227–2237, 2018.
- [PVG<sup>+</sup>11] F. Pedregosa, G. Varoquaux, A. Gramfort, V. Michel, B. Thirion, O. Grisel, M. Blondel, P. Prettenhofer, R. Weiss, V. Dubourg, J. Vanderplas, A. Passos, D. Cournapeau, M. Brucher, M. Perrot, and E. Duchesnay. Scikit-learn: Machine learning in Python. *Journal of Machine Learning Research*, 12:2825–2830, 2011.
- [RNSS] Alec Radford, Karthik Narasimhan, Tim Salimans, and Ilya Sutskever. Improving language understanding by generative pre-training.
- [SPW<sup>+</sup>13] Richard Socher, Alex Perelygin, Jean Wu, Jason Chuang, Christopher D. Manning, Andrew Ng, and Christopher Potts. Recursive deep models for semantic compositionality over a sentiment treebank. In *Proceedings of the 2013 Conference on Empirical Methods in Natural Language Processing*, pages 1631–1642, Seattle, Washington, USA, October 2013. Association for Computational Linguistics.
- [Too12] Michael Toolan. *Narrative: A critical linguistic introduction*. Routledge, 2012.
- [VC12] G Vinodhini and RM Chandrasekaran. Sentiment analysis and opinion mining: a survey. *International Journal*, 2(6):282–292, 2012.
- [VCB14] Andrea Vanzo, Danilo Croce, and Roberto Basili. A context-based model for sentiment analysis in twitter. In *Proceedings of coling 2014, the 25th international conference on computational linguistics: Technical papers*, pages 2345–2354, 2014.
- [VSP<sup>+</sup>17] Ashish Vaswani, Noam Shazeer, Niki Parmar, Jakob Uszkoreit, Llion Jones, Aidan N. Gomez, Lukasz Kaiser, and Illia Polosukhin. Attention is all you need. *CoRR*, abs/1706.03762, 2017.
- [WDS<sup>+</sup>19] Thomas Wolf, Lysandre Debut, Victor Sanh, Julien Chaumond, Clement Delangue, Anthony Moi, Pierric Cistac, Tim Rault, R’emi Louf, Morgan Funtowicz, and Jamie Brew. Huggingface’s transformers: State-of-the-art natural language processing. *ArXiv*, abs/1910.03771, 2019.
- [WHZ<sup>+</sup>16] Yequan Wang, Minlie Huang, Li Zhao, et al. Attention-based lstm for aspect-level sentiment classification. In *Proceedings of the 2016 conference on empirical methods in natural language processing*, pages 606–615, 2016.
- [WWB<sup>+</sup>04] Janyce Wiebe, Theresa Wilson, Rebecca Bruce, Matthew Bell, and Melanie Martin. Learning subjective language. *Computational linguistics*, 30(3):277–308, 2004.
- [ZWX16] Xinjie Zhou, Xiaojun Wan, and Jianguo Xiao. Attention-based lstm network for cross-lingual sentiment classification. In *Proceedings of the 2016 conference on empirical methods in natural language processing*, pages 247–256, 2016.
- [ZZL<sup>+</sup>09] Changli Zhang, Daniel Zeng, Jiexun Li, Fei-Yue Wang, and Wanli Zuo. Sentiment analysis of chinese documents: From sentence to document level. *J. Am. Soc. Inf. Sci. Technol.*, 60(12):2474–2487, December 2009.