Opinion Mining with Deep Contextualized Embeddings

Wen-Bin Han
National Tsing Hua University
HsinChu, 30013, Taiwan, R.O.C.
vincent.han@nlplab.cc

Noriko Kando
National Institute of Informatics
Tokyo, 101-8430, JAPAN
kando@nii.ac.jp

Abstract

Detecting opinion expression is a potential and essential task in opinion mining that can be extended to advanced tasks. In this paper, we considered opinion expression detection as a sequence labeling task and exploited different deep contextualized embedders into the state-of-the-art architecture, composed of bidirectional long short-term memory (BiLSTM) and conditional random field (CRF). Our experimental results show that using different word embeddings can cause contrasting results, and the model can achieve remarkable scores with deep contextualized embeddings. Especially, using BERT embedder can significantly exceed using ELMo embedder.

1 Introduction

One of the crucial tasks in sentiment analysis field is opinion mining, which right now becomes more and more popular for survey. The purpose of opinion mining is to detect the emotional expression from a sentence or an entire document. To be more specific, those expressions usually contain human beings' emotions, interests, even attitudes via natural language.

Fine-grained opinion mining is not only fundamental but also important because bountiful Natural Language Processing (NLP) applications can benefit from it. For example, detecting opinion expression can be extended to identify opinion entity, such as Katiyar and Cardie (2016); Miwa and Bansal (2016); Katiyar and Cardie (2017), recognize stance (Somasundaran and Wiebe, 2010), and extract aspect (Xu et al., 2018; Wang et al., 2016).

Opinion expression detection can be viewed as a linguistic sequence labeling problem. Therefore, recognizing opinionated span from a sentence can be designed as a token-level sequence tagging problem. In this case, standard BIO encoding is usually applied in the same way in İrsoy and Cardie (2014); Choi et al. (2005). Thus, we used the dataset tagged with B, I, and O characters, which represent the beginning, inside, and outside respectively. The first token in each opinionated span is attached to B character, and then the rest of the span are assigned to I character. Table 1 is an example of BIO scheme.

I	hope	you	are	going
O	B_DSE	O	O	O
to	see	more	of	an
O	O	B_ESE	I_ESE	I_ESE
effect	from	this	event	.
I_ESE	O	O	O	O

Table 1: An example with BIO labels for DSE and ESE.

Out of exactness, in this study, we chose the dataset MPQA 1.2 used in Xie (2017); İrsoy and Cardie (2014). To estimate the generalization of our model, we also took another opinion-oriented dataset MOAT from the organization NTCIR7 (Seki et al., 2008).

Opinion expression usually contains a speaker's emotion; hence, we assume that semantics contributes more than syntax. Even though pre-trained word embedding can improve the performance, it still doesn't fully utilize word meaning and its context. Therefore, generating word representations based on the contexts is critical. Owing to deep neural network, model can produce dynamic word representation, such as McCann et al. (2017); Peters et al. (2018); Akbik et al. (2018), on the contrary to fixed representation (Pennington et al., 2014; Mikolov et al., 2013). In this paper, we applied two state-of-the-art and innovative models, ELMo (Peters et al., 2018) and BERT (Devlin et al., 2018), to produce word representations and compared the performances. After obtaining the contextualized word representations,

Proceedings of the 2019 Conference of the North American Chapter of the Association for Computational Linguistics: Student Research Workshop, pages 35–42
Minneapolis, Minnesota, June 3 - 5, 2019.

we fed them into a robust neural network.

2 Related Work

Word Representation. In the past few years, distributed word representations, also known as word embeddings, have been broadly applied to NLP tasks because adding pre-trained one can improve the performance by 1 to 2 point.

Many researches are done with GloVe (Pennington et al., 2014) or Word2Vec (Mikolov et al., 2013). However, one vector can not represent all the meanings of a word. For the sake of deep neural networks, we can add contextual characteristic into word representation during converting. For instance, CoVe (McCann et al., 2017) learns contextualized word vectors via encoders in translation. ELMo (Peters et al., 2018) produces contextualized embeddings from language models computed on 2-layer BiLSTMs with character convolutions. Akbik et al. (2018) trained a character language model to produce contextual string embeddings.

Different from the models mentioned above, BERT (Devlin et al., 2018) extends Radford (2018), extracting features by Transformer (Vaswani et al., 2017), from uni-direction into bi-direction and outperforms impressively in many tasks. In this paper, we took BERT as our embedder because it can generate contextual features. In contrast, ELMo is selected as our baseline embedder.

According to Ruder and Howard (2018), pre-training on a large amount of unlabeled data do improve the model; thus, we used pre-trained BERT and ELMo model and then did fine-tuning with our data.

Neural Network. Recurrent Neural Network (RNN) (Jain and Medsker, 1999) handles variable length input and performs well on NLP tasks, in particular Long Short-Term Momory (LSTM) (Hochreiter and Schmidhuber, 1997). At present, state-of-the-art approaches for sequence labeling typically use bidirectional LSTMs (BiLSTMs) (Schuster and Paliwal, 1997), and a conditional random field (CRF) decoding layer. BiLSTMs can capture not only the previous information but also the following information.

Recently, RNNs have been generally utilized in sequential modeling, such as İrsoy and Cardie (2014), which stacked 3-layer bidirectional vanilla RNNs and the model outperformed the variants of CRF-based methods. Moreover, LSTM has multiple gates allowing the model to learn long-distance dependencies. For sequential labeling tasks, such as Name Entity Recognition (NER) and Part-of-speech (POS) tagging, BiLSTM can take both former and latter contexts into consideration without length limitation and perform better (e.g., Liu et al. (2015) resorted to LSTM with some linguistic features and Katiyar and Cardie (2016) applied deep BiLSTMs on detecting opinion entities and relations).

So far, there have been many variants of LSTM-based neural network models proposed to improve the model and achieve competitive performance against traditional models. Miwa and Bansal (2016) made use of dependency tree and fed it into tree-structured BiLSTM. Simpler and more ingenious, Katiyar and Cardie (2017) only employed attention-based BiLSTM for entity and relation without any manual features or dependency structure information.

Conditional random fields (CRFs) (Lafferty et al., 2001) have also been quite successful for sequence tasks. Thanks to CRF layer, the model can take advantage of sentence-level and neighbor tag information to predict current tag (Huang et al., 2015; Ma and Hovy, 2016).

Some researches combined CNN with RNN to gain the benefits from each other. They extracted character-level features via CNN and handled sentences via RNN, such as Chiu and Nichols (2016). BiLSTM-CNNs-CRF (Ma and Hovy, 2016) obtained features via CNN and stacks CRF on BiLSTM. Xie (2017) applied CNN with bi-direction Gated Recurrent Units (GRUs). Xu et al. (2018) applied CNN to extract features with dual embeddings.

Our study compared BiLSTM-CNNs-CRF (2016) with the model using ELMo as embedder (ELMo-BiLSTM-CRF) and then competed ELMo-BiLSTM-CRF with the one using BERT as embedder (BERT-BiLSTM-CRF). Afterwards, we compared the performance and the difference between two embedders.

Eventually, based on Katiyar and Cardie (2017); Vaswani et al. (2017), we employed attention mechanism in BERT-BiLSTM-CRF layer expected to improve the accuracy.

3 Model

The main focus of our research is to improve the performance by using the deep contextualized word representation and compare two kinds of word embedders. Therefore, rather than static word representations, we employed dynamic word embedders that create the word representation based on its context.

After transforming all tokens into continuous vector representations, we fed them into BiLSTM instead of feed-forward network because BiLSTM neural network is prevailing and dominant on NLP tasks and many opinion-related tasks are completed with it.

After finishing each epoch, we updated the parameters simultaneously via back-propagation through time (BPTT) (Werbos, 1990). Last, after our comparisons, we chose the better model and added attention mechanism on BiLSTM layer to strengthen the performance.

In the following subsections, we decompose our neural network architectures and describe the components (layers) in detail. Hence, we introduce the neural layers in our models one-by-one from bottom to top. Before describing the models, we first illustrate the annotation format of data, which is followed by the most important part, word embedders. Afterwards, the BiLSTM layers as well as CRF layer will be mentioned.

3.1 Data Scheme

Opinion expression detection can be viewed as a linguistic sequence labeling problem. In this case, BIO encoding is usually applied to identify the opinionated span in a sentence. Thus, in the dataset, each token is tagged with BIO.

3.2 Word Embedders

According to previous research, using pre-trained word embeddings in downstream tasks usually outperforms using randomly initialized vectors. Therefore, choosing a robust embedding way to transform tokens is influential in experiments. Nevertheless, although there have been abundant ways to convert words into dense distributions so far, we first did some experiments to discover how effective each approach is.

After observing the datasets, we found that opinion detection task concentrates more on semantics than syntax. Consequently, we used a deep neural network, BERT, to figure out this problem because it is flexible enough to produce the representation of each token based on its context, even more the whole sentence.

Moreover, the BERT model is so overwhelming that it works impressively on plenty of tasks. In order to examine the performance on different embedding in opinion mining, we used BERT model as the word embedder in our experiments. Besides, it is necessary to compare the main model with a baseline. Therefore, we took another contextualized embedding, ELMo, as word embedding, and regarded the results as our baseline.

3.3 Architecture

We stacked BiLSTM on top of embedders because of the following reasons. First of all, in our research, BERT and ELMo are only used as word embedders instead of the whole architecture. Second, many RNN-based neural network models are proposed to figure out the sequence labeling tasks, and also achieved competitive performance against traditional models (Ma and Hovy, 2016; Huang et al., 2015). Last but not least, we would like to compare the performances between different contextualized embeddings so we must fix the other parts of architecture.

In addition, we also added CRF (Lafferty et al., 2001) layer because it can consider the correlations between labels in neighborhoods to predict current tag. Thanks to CRF layer, the model took full advantage of sentence-level tag information and enhanced itself to decode the best chain of labels for a given input sentence. In summary, we combined BERT or ELMo embedders with BiLSTM and CRF layers (Bert-BiLSTM-CRF and ELMo-BiLSTM-CRF). Each model can efficiently benefit from the forward and backward input features through BiLSTM layer and sentence-level tag information via CRF layer.

3.4 Attention

Katiyar and Cardie (2017) displayed that attention mechanism can reinforce the model. Therefore, we extended the Bert-BiLSTM-CRF network with multi-head attention approach because it allows the model to jointly attend to information from different representation sub-spaces at different positions (Vaswani et al., 2017). More concretely, the hidden states from BiLSTM layers went through attention layer and then linear layer as well as CRF layer.

Dataset	Count
DSE	14492
ESE	14492
NTCIR7-MOAT	3376

Table 2: The number of sentences for each dataset.

4 Experiments

4.1 Data Sets

Due to the development of opinion mining, there are numerous existing datasets which are useful for us. For example, Multi-Perspective Question Answering (MPQA) offers the annotated dataset.

In this paper, we conducted the experiments on the processed dataset, MPQA 1.2, provided by İrsoy and Cardie (2014). It includes two types of opinion expression proposed by Wiebe et al. (2005) - direct subjective expressions(DSEs) and expressive subjective expressions (ESEs). DSEs represent both subjective speech events and explicitly mentioned private states, while ESEs consist of tokens that express emotion or sentiment in an indirect or implicit way. Table 1 is also an example of DSE and ESE.

In addition to MQPA 1.2, we have another dataset from MOAT task, which is also related to opinion expression and organized by NTCIR (Seki et al., 2008). Different datasets can confirm that our model is generalized enough. Table 2 depicts the number of sentences in each dataset.

In the research, we first used one tenth of the dataset as the development set, and the rest of the data are applied 10-fold in order to get a more balanced result.

4.2 Word Embeddings

This experiment contains the details of the comparison between ELMo and other embedding ways. Table 3 is the results from different notable embedding ways on DSEs, and the evaluation is token-based calculation.

we selected some pre-trained word embedding - Word2Vec (Mikolov et al., 2013), GloVe (Pennington et al., 2014), and dependency-based word embedding (Levy and Goldberg, 2014). Besides, we also tried BiLSTM-CNNs-CRF (Ma and Hovy, 2016), which extracted character features by CNN and concatenated GloVe.

All the architectures are identical except for the word representings and decoding ways. Each combination is composed of one type of embeddings and three BiLSTM layers with or without CRF layer.

According to the results, the model with CRF layer does defeat the model without CRF layer. It proves that adding CRF layer becomes more powerful. Moreover, all the scores are close except for ELMo one. BiLSTM-CNNs-CRF doesn't exceed the other models only with pre-trained word embedding. We assumed that opinion mining emphasizes more semantics than syntax. Therefore, character features do not boost the model much.

Subsequently, we compared BiLSTM-CNNs-CRF with the one using ELMo embedder (ELMo-BiLSTM-CRF), and it turned out that ELMo-BiLSTM-CRF surpasses BiLSTM-CNNs-CRF substantially. Therefore, we took it as our baseline in this paper.

4.3 Parameter Settings

We made use of pre-trained BERT and ELMo models and did the fine-tuning during training procedure guided by Devlin et al. (2018) in order to make the model learn the distribution of the dataset.

For ELMo model, we used ELMoForManyLangs [1] provided by HIT-SCIR (Che et al., 2018; Fares et al., 2017) and took the average of the hidden states from all the layers as token representation.

For BERT model, we used "BERT-Base, Multilingual Cased" provided by Google [2] (Devlin et al., 2018). Most of the hyper-parameters are instructed by the paper. Only the hidden state from the last layer is picked and regarded as token representation.

For the rest of shared properties, we stacked 3 layers of BiLSTM [3] based on İrsoy and Cardie (2014). Learning rate is 0.00005 advised by (Devlin et al., 2018). Dropout is set to 0.5 before fully connected layer. Optimizer uses Adam (Kingma and Ba, 2014).

4.4 Evaluation

In the evaluation, we calculated precision, recall, and F1-score. However, in order to evaluate our model comprehensively, we measured our model not only by token basis but also span basis. This

[1] https://github.com/HIT-SCIR/ELMoForManyLangs

[2] https://github.com/google-research/bert

[3] According to our experiments, the number of layers do not affect the result much.

Embedding	Non CRF		CRF	
	Dev	**Test**	**Dev**	**Test**
GloVe	0.5249	0.5483	0.5474	0.5546
Word2Vec	0.5363	0.5387	0.5347	0.5685
Dependency-based word embedding	0.5224	0.5407	0.5450	0.5630
CNN character embedding and Glove	0.5395	0.5339	0.5455	0.5686
ELMo	**0.5920**	**0.5928**	**0.6151**	**0.6293**

Table 3: The results of applying a variety of embeddings on DSEs. ELMo embedder outperforms others significantly. Besides, stacking a CRF layer as decoder can increase the performance slightly.

Model	Token basis			Binary Overlap			Proportional Overlap		
	Prec.	Rec.	F1	Prec.	Rec.	F1	Prec.	Rec.	F1
ELMo-BiLSTM-CRF	0.708	0.640	0.640	0.703	0.620	0.653	0.676	0.543	0.597
BERT-BiLSTM-CRF	0.735	0.753	0.720	0.738	0.766	0.750	0.702	0.705*	0.701
BERT-BiLSTM-Attn-CRF	**0.740**	**0.761**	**0.723**	**0.744**	**0.768**	**0.752**	**0.710**	0.705*	**0.703**

Table 4: Experimental evaluation of models for DSE.

Model	Token basis			Binary Overlap			Proportional Overlap		
	Prec.	Rec.	F1	Prec.	Rec.	F1	Prec.	Rec.	F1
ELMo-BiLSTM-CRF	0.637	0.518	0.552	0.644	0.510	0.560	0.586	0.391	0.460
BERT-BiLSTM-CRF	**0.672**	0.608	0.631	**0.686**	0.634	0.654	0.625	0.527	0.564
BERT-BiLSTM-Attn-CRF	0.665	**0.635**	**0.645**	0.654	**0.684**	**0.663**	0.598	**0.570**	**0.577**

Table 5: Experimental evaluation of models for ESE.

Model	Token basis			Binary Overlap			Proportional Overlap		
	Prec.	Rec.	F1	Prec.	Rec.	F1	Prec.	Rec.	F1
ELMo-BiLSTM-CRF	0.483	0.419	0.404	**0.465**	0.356	0.323	**0.426**	0.267	0.245
BERT-BiLSTM-CRF	0.468	0.428	0.430	0.426	0.432	**0.415**	0.387	0.354	0.359*
BERT-BiLSTM-Attn-CRF	**0.515**	**0.481**	**0.482**	0.360	**0.510**	0.407	0.337	**0.427**	0.359*

Table 6: Experimental evaluation of models for NTCIR-MOAT7. * means the same scores.

is because it is difficult to define the boundaries of expressions, even for human annotators. To put it another way, for token-based evaluation, F1-score pays attention to whether the individual tag is predicted correctly or not. In contrast, span-based evaluation cares about the count of overlap; hence, we used binary overlap along with proportional overlap. Binary overlap computes the number of matching overlaps between predicted sequence and ground-truth sequence. As long as predicted span overlaps ground-truth span, binary overlap views it as a correct prediction. To refine the evaluation, proportional overlap considers the length of overlap and imparts a partial correctness to each match, which is able to assess the model more accurate. We used these three evaluations to measure our models.

5 Results and Discussion

5.1 Results

Table 4, 5, and 6 display the results of the data sets individually. The scores show that using BERT embedder does achieve better performance than using ELMo embedder by dramatic difference, which implies that BERT is promising in opinion mining tasks. More interestingly, adding attention mechanism on BiLSTM can increase the performance slightly.

5.2 Discussion

According to the scores, it is easier to detect DSEs than ESEs because DSEs contain clear opinion-

DSE S1	My public affairs keepers [could]$_B$ [n't care less]$_I$.
ELMo-BiLSTM-CRF	My public affairs keepers could n't care less .
Bert-BiLSTM-CRF	My public affairs keepers [could]$_B$ [n't care]$_I$ less .
Bert-BiLSTM-Attn-CRF	My public affairs keepers [could]$_B$ [n't care less]$_I$.
ESE S1	By comparison , the al Qaedans [look]$_B$ [pretty fat , if not happy]$_I$.
ELMo-BiLSTM-CRF	By comparison , the al Qaedans look [pretty]$_B$ [fat]$_I$, [if]$_B$ [not happy]$_I$.
Bert-BiLSTM-CRF	[By]$_B$ comparison , the al Qaedans [look]$_B$ [pretty fat]$_I$, [if]$_B$ [not happy]$_I$.
Bert-BiLSTM-Attn-CRF	[By]$_B$ [comparison]$_I$, the al Qaedans [look]$_B$ [pretty fat]$_I$, [if]$_B$ [not happy]$_I$.
ESE S2	Their restroom arrangements [are]$_B$ [pretty spartan]$_I$.
ELMo-BiLSTM-CRF	Their restroom arrangements are [pretty]$_B$ [spartan]$_I$.
Bert-BiLSTM-CRF	Their restroom arrangements are [pretty]$_B$ [spartan]$_I$.
Bert-BiLSTM-Attn-CRF	Their restroom arrangements are [pretty]$_B$ [spartan]$_I$.
ESE S3	We can see for ourselves , [sort]$_B$ [of]$_I$.
ELMo-BiLSTM-CRF	We can see for ourselves , [sort]$_I$ of .
Bert-BiLSTM-CRF	We can [see]$_I$ for [ourselves]$_I$, [sort]$_B$ [of]$_I$.
Bert-BiLSTM-Attn-CRF	We can see for ourselves [,]$_B$ [sort]$_I$ [of]$_I$.
MOAT S1	[He]$_B$ [considers them as Indonesian as he is .]$_I$
ELMo-BiLSTM-CRF	He considers [them]$_I$ as [Indonesian]$_I$ as [he]$_I$ is .
Bert-BiLSTM-CRF	He considers them as [Indonesian]$_I$ as he is .
Bert-BiLSTM-Attn-CRF	[He]$_B$ [considers them as Indonesian as he is .]$_I$

Table 7: Output from our models for each dataset.

ated expression, which may be some adjectives. However, detecting ESEs requires understanding more implicit semantics, but BERT-BiLSTM-CRF works well on ESEs.

Adding attention mechanism does not improve much probably because BERT layer has already incorporated multi-head attention mechanism and caught well-represented information.

Moreover, the training epochs for ELMo-BiLSTM-CRF is 4 times more than that of BERT-BiLSTM-CRF to converge. In other words, applying BERT embedder can save much more time. To conclude, in our experiments, BERT embedder is much more efficient than other embeddings.

5.3 Error Analysis

In this section, we observed the predictions and analyzed the defect in our models. Table 7 is our several predictions from our model on each dataset. Many sentences are predicted approximately or even same; however, in some sentences ELMo-BiLSTM-CRF has lower recall.

BERT-BiLSTM-CRF can predict well, but some failures are caused by the inconsistency in dataset. For example, whether definite articles (e.g. "the") or punctuation should be included or not is one of the problems. Besides, the same verb, such as 'say', in similar contexts is not always annotated, either.

For ESEs, It is much more difficult to clearly identify implicit semantics because there are many fragmented predictions. Besides, although we have CRF layer to consider the entire sequence predictions, it still exists some wrong tagging, such as starting with I tag.

The opinionated spans in NTCIR7-MOAT data are usually too long, which is a little different from the other datasets. Besides, the number of sentences is not much; thus, the result does not meet the expectation.

Once the flaws in dataset are figured out, we can gain a better performance. Furthermore, adding another feature, such as GLoVe (Pennington et al., 2014) or linguistic characteristics, is also a way to enhance the model.

6 Conclusion

In this paper, we introduced contextualized embeddings into opinion mining task. Experimentally, our models have significant promotion for changing embedder and prove that deep contextualized embeddings perform well in opinion mining task. Specifically, our comparison shows that using BERT embedder dramatically surpasses using ELMo embedder.

In the future work, it would be better to supplement other word embedding (Pennington et al., 2014; Mikolov et al., 2013) as auxiliary, just like

Chiu and Nichols (2016); Xu et al. (2018). Even more, we can add contextual string embedding (Akbik et al., 2018) to support character-level features and apply it to advanced opinion mining tasks.

Acknowledgments

Thanks to the professor Noriko Kando to supervise me for the research and my friend for giving the advice for the paper. This work was supported by the International Internship Program of National Institute of Informatics, Japan, and JSPS KAKENHI Grant Numbers JP18H03338 and JP16H01756.

References

Alan Akbik, Duncan Blythe, and Roland Vollgraf. 2018. Contextual string embeddings for sequence labeling. In *COLING*.

Wanxiang Che, Yijia Liu, Yuxuan Wang, Bo Zheng, and Ting Liu. 2018. Towards better UD parsing: Deep contextualized word embeddings, ensemble, and treebank concatenation. In *Proceedings of the CoNLL 2018 Shared Task: Multilingual Parsing from Raw Text to Universal Dependencies*, pages 55–64, Brussels, Belgium. Association for Computational Linguistics.

Jason P. C. Chiu and Eric Nichols. 2016. Named entity recognition with bidirectional lstm-cnns. *Transactions of the Association for Computational Linguistics*, 4:357–370.

Yejin Choi, Claire Cardie, Ellen Riloff, and Siddharth Patwardhan. 2005. Identifying sources of opinions with conditional random fields and extraction patterns. In *HLT/EMNLP*.

Jacob Devlin, Ming-Wei Chang, Kenton Lee, and Kristina Toutanova. 2018. Bert: Pre-training of deep bidirectional transformers for language understanding. *CoRR*, abs/1810.04805.

Murhaf Fares, Andrey Kutuzov, Stephan Oepen, and Erik Velldal. 2017. Word vectors, reuse, and replicability: Towards a community repository of large-text resources. In *Proceedings of the 21st Nordic Conference on Computational Linguistics*, pages 271–276, Gothenburg, Sweden. Association for Computational Linguistics.

Sepp Hochreiter and Jürgen Schmidhuber. 1997. Long short-term memory. *Neural Comput.*, 9(8):1735–1780.

Zhiheng Huang, Wei Xu, and Kai Yu. 2015. Bidirectional lstm-crf models for sequence tagging. *CoRR*, abs/1508.01991.

Ozan İrsoy and Claire Cardie. 2014. Opinion mining with deep recurrent neural networks. In *Proceedings of the Conference on Empirical Methods in Natural Language Processing*, pages 720–728.

L. C. Jain and L. R. Medsker. 1999. *Recurrent Neural Networks: Design and Applications*, 1st edition. CRC Press, Inc., Boca Raton, FL, USA.

Arzoo Katiyar and Claire Cardie. 2016. Investigating lstms for joint extraction of opinion entities and relations. In *ACL*.

Arzoo Katiyar and Claire Cardie. 2017. Going out on a limb: Joint extraction of entity mentions and relations without dependency trees. In *ACL*.

Diederik P. Kingma and Jimmy Ba. 2014. Adam: A method for stochastic optimization. *CoRR*, abs/1412.6980.

John D. Lafferty, Andrew McCallum, and Fernando Pereira. 2001. Conditional random fields: Probabilistic models for segmenting and labeling sequence data. In *ICML*.

Omer Levy and Yoav Goldberg. 2014. Dependency-based word embeddings. In *ACL*.

Pengfei Liu, Shafiq R. Joty, and Helen M. Meng. 2015. Fine-grained opinion mining with recurrent neural networks and word embeddings. In *EMNLP*.

Xuezhe Ma and Eduard H. Hovy. 2016. End-to-end sequence labeling via bi-directional lstm-cnns-crf. *CoRR*, abs/1603.01354.

Bryan McCann, James Bradbury, Caiming Xiong, and Richard Socher. 2017. Learned in translation: Contextualized word vectors. In *NIPS*.

Tomas Mikolov, Ilya Sutskever, Kai Chen, Gregory S. Corrado, and Jeffrey Dean. 2013. Distributed representations of words and phrases and their compositionality. In *NIPS*.

Makoto Miwa and Mohit Bansal. 2016. End-to-end relation extraction using lstms on sequences and tree structures. *CoRR*, abs/1601.00770.

Jeffrey Pennington, Richard Socher, and Christopher D. Manning. 2014. Glove: Global vectors for word representation. In *EMNLP*.

Matthew E. Peters, Mark Neumann, Mohit Iyyer, Matt Gardner, Christopher Clark, Kenton Lee, and Luke S. Zettlemoyer. 2018. Deep contextualized word representations. In *NAACL 2018*.

Alec Radford. 2018. Improving language understanding by generative pre-training.

Sebastian Ruder and Jeremy Howard. 2018. Universal language model fine-tuning for text classification. In *ACL*.

M. Schuster and K.K. Paliwal. 1997. Bidirectional recurrent neural networks. *Trans. Sig. Proc.*, 45(11):2673–2681.

Yohei Seki, David Kirk Evans, Lun-Wei Ku, Le Sun, Hsin-Hsi Chen, and Noriko Kando. 2008. Overview of multilingual opinion analysis task at ntcir-7. *Proceedings of The IEEE - PIEEE*.

Swapna Somasundaran and Janyce Wiebe. 2010. Recognizing stances in ideological on-line debates.

Ashish Vaswani, Noam Shazeer, Niki Parmar, Jakob Uszkoreit, Llion Jones, Aidan N. Gomez, Lukasz Kaiser, and Illia Polosukhin. 2017. Attention is all you need. In *NIPS*.

Xin Wang, Yuanchao Liu, Chengjie Sun, Ming Liu, and Xiaolong Wang. 2016. Extended dependency-based word embeddings for aspect extraction. In *ICONIP*.

Janyce Wiebe, Theresa Wilson, and Claire Cardie. 2005. Annotating expressions of opinions and emotions in language. *Language Resources and Evaluation*, 39(2):165–210.

Xiaoxia Xie. 2017. Opinion expression detection via deep bidirectional c-grus. *2017 28th International Workshop on Database and Expert Systems Applications (DEXA)*, pages 118–122.

Hu Xu, Bing Liu, Lei Shu, and Philip S. Yu. 2018. Dual embeddings and cnn-based sequence labeling for aspect extraction. In *ACL 2018*.