NLP Seminar Report

Agajan Torayev
MSc Student / Rheinische Friedrich-Wilhelms-Universitt Bonn
s6agtora@uni-bonn.de

Abstract

Text classification is one of fundamental tasks in Natural Language Processing. Since it is used in many applications such as sentiment analysis, information retrieval, legal and medical document classification, research in text classification has attracted attention of many researchers in NLP. In this seminar I have chosen topic of text classification using artificial neural networks, particularly convolutional neural networks and recurrent neural networks, since recent advances in Deep Learning show big improvements in NLP tasks compared to other Machine Learning algorithms.

1 Papers

For this seminar talk I have chosen 3 papers:

1. Initializing Convolutional Filters with Semantic Features for Text Classification (Li et al., 2017)

2. Sequential Short-Text Classification with Recurrent and Convolutional Neural Networks (Lee and Dernoncourt, 2016)

3. Recurrent Convolutional Neural Networks for Text Classification (Lai et al., 2015)

2 Introduction

All the papers I have chosen discuss text classification task using neural networks but with different perspective. Each paper addresses different problem we face when performing text classification task and using neural networks. Authors present their papers by giving introduction to the problem they are addressing in the paper, they give basic intuition for the solution of the problem, after that they thoroughly discuss method or model they came up with, and as a final part they give experiment results they performed.

In this seminar report I am not going to discuss every paper separately, instead I have tried to summarize all of them according to the following sections:

- Key problems papers address.
- Intuition behind proposed methods and models.
- Details of models and methods.
- Experiments and Results.
- Conclusion.

3 Key problems papers address

All of the papers I have chosen discuss text classification using neural networks, but they address different problems. Paper "Initializing Convolutional Filters with Semantic Features for Text Classification" by Li (2017) discuss very important problem of using neural networks in text classification - weight initialization. The objective function of neural networks is non-convex and thus neural networks are very sensitive to weight initialization. To solve this problem authors propose initializing convolutional filters with semantic features using n-grams extracted from documents. Paper "Sequential Short-Text Classification with Recurrent and Convolutional" by Lee and Dernoncourt (2016) address a problem of short text classification task in isolation, they give arguments why short texts should be considered using preceding short texts in classification task. Paper " Recurrent Convolutional Neural Networks for Text Classification" by Lai (2015) discuss about feature representation problem in text

classifcation, and author proposes using Recurrent Convolutional Neural Networks to capture contextual information and extract important features automatically.

4 Intuition behind proposed methods and models

The ideas for the solutions of problems are very intuitive and they give an overview of the models. For example for solving weight initialization Li (2017) proposes to use n-grams extracted from the document and use them to initialize convolutional filters. For the problem of isolated short text classfication Lee and Dernoncourt (2016) propose to use short text representation using either RNN or CNN and feed them to 2-layer feed forward network with some preceding short texts. For capturing contextual information in Convolutional Neural Networks Lai(2015) suggests bidirectional LSTM with max pooling layer.

5 Details of models and methods

In this section I am going to discuss methods and models more in details.

5.1 Initializing Convolutional Filters with Semantic Features for Text Classification

For weight initialization in Convolutional Neural Networks authors propose to use extract n-grams from documents and use them to initialize convolutional filters. Model consists of 2 steps: **n-gram selection** and **filter initialization**. To select important n-grams from the documents Naive Bayes is used to determine the word's importance in document using the given formula:

$$r = \frac{(p_c^w + \alpha)/\|p_c\|_1}{(p_{\tilde{c}}^w + \alpha)/\|p_{\tilde{c}}\|_1} \quad (1)$$

Where p_c^w is the number of texts that contain n-gram w in class c; $p_{\tilde{c}}^w$ is the number of texts that contain n-gram w in other classes; $\|p_c\|_1$ is the number of text in class c; $\|p_{\tilde{c}}\|_1$ is the number of texts in other classes; and α is a smoothing parameter.

For example, given in the paper, for positive class in movie review dataset the ratios the ratios of n-grams like amazing and not bad should belarge since they appear much more frequently, while for neutral n-grams like of the and movie, their ratios shouldbe around 1. Thus, for each class, we select n-grams whose ratios are much higher than 1 for filter initialization. After n-grams have been extracted, word embeddings are concatenated to construct n-gram embeddings. In filter initialization step n-gram embeddings are used to initialize filters, but instead of directly using n-gram embeddings K-means is used to cluster features of n-grams and the clusters' centroid vectors are fed to the center of filters, remaining positions still being randomly initialized.

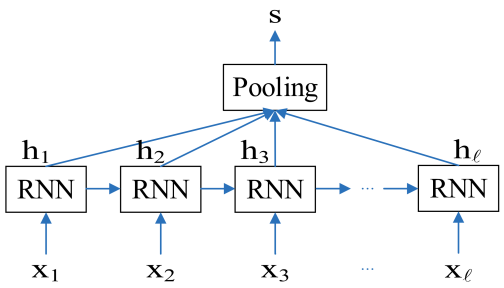

Figure 1: RNN-based short text representation for generating the vector representation s of a short text $x_{1:l}$ (Lee and Dernoncourt, 2016)

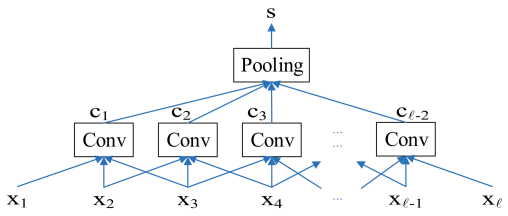

Figure 2: CNN-based short text representation for generating the vector representation s of a short text $x_{1:l}$. Conv refers to convolution operations, and the filter height $h = 3$ is used in this figure.(Lee and Dernoncourt, 2016)

5.2 Sequential Short-Text Classification with Recurrent and Convolutional Neural Networks

In this paper the problem of isolated short text classification is solved in two main steps. In the first step short text representation is learned either using RNN (Figure 1) or CNN (Figure 2).

In the second step learned short text representation is fed into two-layer feedforward neural network. The first layer takes as input $\mathbf{s}_{i-d_1-d_2:i}$ and outputs the sequence $y_{i-d_2:i}$ and the second layer takes as input the sequence of class representations $y_{i-d_2:i}$ and outputs $\mathbf{z}_i \in \mathbb{R}^k$ (the probability distribution over the set of k classes for the i^{th} short-ext). Here, d_1, d_2 are the history sizes (hyperpa-

rameters) used in the first and second layers, respectively.

5.3 Recurrent Convolutional Neural Networks for Text Classification

This paper addresses the problem of capturing contextual information to the greatest possible extent and automatic feature representation learning. Two key components of the model are bi-directional recurrent structure and max-pooling layer that automatically judges which features play key roles in text classification. The model consists of two parts: word representation learning and text representation learning. In word representation learning left and right contexts of word w_i is learned in the following way:

$$c_l(w_i) = f(W^{(l)}c_l(w_{i-1}) + W^{(sl)}e(w_{i-1})) \quad (2)$$

$$c_r(w_i) = f(W^{(r)}c_r(w_{i+1}) + W^{(sr)}e(w_{i+1})) \quad (3)$$

Where $c_l(w_i)$ is left context of word w_i; $c_r(w_i)$ is right context of word w_i; $e(w_{i-1})$ and $e(w_{i+1})$ are the word embeddings of words w_{i-1} and w_{i+1}; $W^{(l)}$ is a matrix that transforms the hidden layer (context) into the next hidden layer; $W^{(sl)}$ is a matrix used to combine the semantic of the current word with the next word's left context; f is a non-linear activation function. Left and right contexts of word are learned because they can capture the semantics of all left- and right-side contexts.c_l, $e(w_i)$, c_r are concatenated to obtain the representation for the word w_i:

$$x_i = [c_l(w_i); e(w_i); c_r(w_i)] \quad (4)$$

After the representation x_i is obtained, linear transformation function with $tanh$ activation function is applied to get latent semantic vector $\mathbf{y}_i^{(2)} = tanh(W^{(2)}\mathbf{x}_i + \mathbf{b}^{(2)})$. When all of the representations of words are calculated, a max-pooling layer is applied to obtain fixed-length vector $\mathbf{y}^{(3)} = \max_{i=1}^{n}(\mathbf{y}_i^{(2)})$. The last part is an output layer: $\mathbf{y}^{(4)} = W^{(4)}\mathbf{y}^{(3)} + \mathbf{b}^{(4)}$ and finally, the softmax function is applied to $\mathbf{y}^{(4)}$.

6 Experiments and Results

Paper by (Li et al., 2017) uses the following datasets: MR (Pang and Lee, 2005), SST-1/2 (Socher et al., 2013), Subj (Pang and Lee, 2004),

Model	MR	SST-1	SST-2	Subj	TREC	CR	MPQA
CNN-non-static (Kim, 2014)	81.5	48.0	87.2	93.4	93.6	84.3	**89.5**
MV-CNN (Yin and Schütze, 2016)	-	49.6	**89.4**	93.9	-	-	-
MGNC-CNN (Zhang et al., 2016b)	-	48.7	88.3	**94.1**	**95.5**	-	-
CNN-Rule (Hu et al., 2016)	81.7	-	89.3	-	-	85.3	-
Our Model (CNN-non-static+UNI)	**82.1**	**50.8**	89.0	93.7	94.4	**86.0**	89.3
combine-skip (Kiros et al., 2015)	76.5	-	-	93.6	92.2	80.1	87.1
Adasent (Zhao et al., 2015)	**83.1**	-	-	**95.5**	92.4	**86.3**	**93.3**
DSCNN (Zhang et al., 2016a)	82.2	50.6	**88.7**	93.9	**95.6**	-	-
PV (Le and Mikolov, 2014)	74.8	48.7	87.8	90.5	91.8	78.1	74.2
NBSVM (Wang and Manning, 2012)	79.4	-	-	93.2	-	81.8	86.3
Tree LSTM (Tai et al., 2015)	-	**51.0**	88.0	-	-	-	-

Figure 3: Comparisons of the state of the arts for paper by (Li et al., 2017)

Model	DSTC 4	MRDA	SwDA
CNN	65.5	**84.6**	**73.1**
LSTM	**66.2**	84.3	69.6
Majority class	25.8	59.1	33.7
SVM	57.0	–	–
Graphical model	–	81.3	–
Naive Bayes	–	82.0	–
HMM	–	–	71.0
Memory-based Learning	–	–	72.3
Interlabeler agreement	–	–	84.0

Figure 4: Comparison of SoTA for paper by (Lee and Dernoncourt, 2016)

TREC (Li and Roth, 2002), CR (Hu and Liu, 2004), and MPQA (Wiebe et al., 2005). Experiments show that n-gram features make a great contribution to both two-class and multi-class classification task. The result of experiments are summarized in Figure 3.

Paper by (Lee and Dernoncourt, 2016) evaluates model on dialog act classification task. This paper uses the following datasets: DSTC 4 (Kim et al., 2016), MRDA (Janin et al., 2003), SwDA (Jurafsky et al., 1997). CNN-based representation model outperforms in MRDA and SwDA datasets, where RNN-based representation model outperforms in DSTC 4 dataset. Overall, model proposed outperforms all previous models in all of 3 datasets. The summary of results is shown in figure 4.

Paper by (Lai et al., 2015) uses 4 datasets to evaluate their model: **20Newsgroups**, **Fudan set**, **ACL Anthology Network**, **Stanford Sentiment Treebank**.The most interesting comparison for this paper is comparison of RCNN and CNN. Authors compared RCNN vs. CNN with different window sizes (1 to 19). In all cases RCNN outperforms CNN with big margin. The result of this comparison is in Figure 5

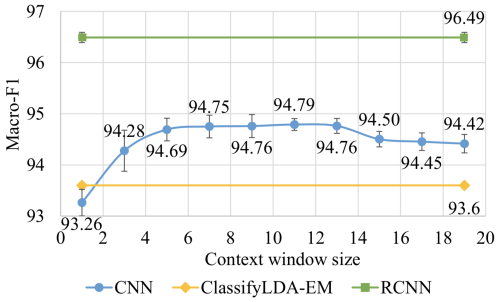

Figure 5: Macro-F1 curve for how context window size influences the performance of the 20Newsgroups classification for paper (Lai et al., 2015)

7 Conclusion

In this seminar I have analyzed 3 different papers from the perspective of text classification. All the papers use neural networks, which confirms the success of Deep Learning algorithms. Papers thoroughly discuss about different problems when using Deep Learning methods for the text classification task. Papers are provided with technical implementation details and also with experiments and results.

Acknowledgments

I want to thank Mohnish Dubey for supervising me and giving advices during the semester. Also Diego Esteves for advising in NLP Lab sessions.

References

Minqing Hu and Bing Liu. 2004. Mining and summarizing customer reviews. In *Proceedings of the Tenth ACM SIGKDD International Conference on Knowledge Discovery and Data Mining*, KDD '04, pages 168–177, New York, NY, USA. ACM.

Adam Janin, Don Baron, Jane Edwards, Dan Ellis, David Gelbart, Nelson Morgan, Barbara Peskin, Thilo Pfau, Elizabeth Shriberg, Andreas Stolcke, and Chuck Wooters. 2003. The icsi meeting corpus. pages 364–367.

Dan Jurafsky, Elizabeth Shriberg, and Debra Biasca. 1997. Switchboard swbd-damsl shallow-discourse-function annotation coders manual. *Institute of Cognitive Science Technical Report*, pages 97–102.

Seokhwan Kim, Luis Fernando D'Haro, Rafael E. Banchs, Jason D. Williams, and Matthew Henderson. 2016. The fourth dialog state tracking challenge. In *IWSDS*, volume 427 of *Lecture Notes in Electrical Engineering*, pages 435–449. Springer.

Siwei Lai, Liheng Xu, Kang Liu, and Jun Zhao. 2015. Recurrent convolutional neural networks for text classification. In *Proceedings of the Twenty-Ninth AAAI Conference on Artificial Intelligence*, AAAI'15, pages 2267–2273. AAAI Press.

Ji Young Lee and Franck Dernoncourt. 2016. Sequential short-text classification with recurrent and convolutional neural networks. *CoRR*, abs/1603.03827.

Shen Li, Zhe Zhao, Tao Liu, Renfen Hu, and Xiaoyong Du. 2017. Initializing convolutional filters with semantic features for text classification. In *Proceedings of the 2017 Conference on Empirical Methods in Natural Language Processing*, pages 1884–1889. Association for Computational Linguistics.

Xin Li and Dan Roth. 2002. Learning question classifiers. In *Proceedings of the 19th International Conference on Computational Linguistics - Volume 1*, COLING '02, pages 1–7, Stroudsburg, PA, USA. Association for Computational Linguistics.

Bo Pang and Lillian Lee. 2004. A sentimental education: Sentiment analysis using subjectivity summarization based on minimum cuts. In *Proceedings of the 42Nd Annual Meeting on Association for Computational Linguistics*, ACL '04, Stroudsburg, PA, USA. Association for Computational Linguistics.

Bo Pang and Lillian Lee. 2005. Seeing stars: Exploiting class relationships for sentiment categorization with respect to rating scales. In *Proceedings of the 43rd Annual Meeting on Association for Computational Linguistics*, ACL '05, pages 115–124, Stroudsburg, PA, USA. Association for Computational Linguistics.

Richard Socher, Alex Perelygin, Jean Wu, Jason Chuang, Christopher D. Manning, Andrew Ng, and Christopher Potts. 2013. 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. Association for Computational Linguistics.

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