AttnConvnet at SemEval-2018 Task 1: Attention-based Convolutional Neural Networks for Multi-label Emotion Classification

Yanghoon Kim[1,2], Hwanhee Lee[1] and Kyomin Jung[1,2]
[1]Seoul National University, Seoul, Korea
[2]Automation and Systems Research Institute, Seoul National University, Seoul, Korea
{ad26kr,wanted1007,kjung}@snu.ac.kr

Abstract

In this paper, we propose an attention-based classifier that predicts multiple emotions of a given sentence. Our model imitates human's two-step procedure of sentence understanding and it can effectively represent and classify sentences. With emoji-to-meaning preprocessing and extra lexicon utilization, we further improve the model performance. We train and evaluate our model with data provided by SemEval-2018 task 1-5, each sentence of which has several labels among 11 given emotions. Our model achieves 5th/1st rank in English/Spanish respectively.

1 Introduction

Since the revolution in deep neural networks, especially with the help of Long short-term memory(Hochreiter and Schmidhuber, 1997), it has been easy for machines to imitate human's linguistic activities, such as sentence classification(Kim, 2014), language model(Sundermeyer et al., 2010), machine translation(Bahdanau et al., 2015).

Emotion classification is a subpart of sentence classification that predicts the emotion of the given sentence by understanding the meaning of it. Multi-label emotion classification requires more powerful ability to comprehend the sentence in variety of aspects. For example, given a sentence 'For real? Look what I got for my birthday present!!', it is easy for human to figure out that the sentence not only expressing 'joy' but also 'surprise'. However, machines may require more task-specific structure to solve the same problem.

Attention mechanisms are one of the most spotlighted trends in deep learning and recently made their way into NLP. Applied to systems with neural networks, it functions as visual attention mechanisms found in humans(Denil et al., 2012) and the most effective region of features will be highlighted over time, making the system better exploit the features related to the training objective. (Bahdanau et al., 2015) is one of the most significant footprints of attention mechanism in NLP and they applied attention mechanisms to machine translation for the first time. The model generates target word under the influence of related source words. Furthermore, Vaswani et al. (2017) proposed a brand new architecture for neural machine translation. The model utilizes attention mechanisms not only as the submodule but also as the main structure, improving time complexity and performance.

Inspired by (Vaswani et al., 2017), we come up with attention-based multi-label sentence classifier that can effectively represent and classify sentences. Our system is composed of a self-attention module and multiple CNNs enabling it to imitate human's two-step procedure of analyzing sentences: comprehend and classify. Furthermore, our emoji-to-meaning preprocessing and extra lexicon utilization improve model performance on given dataset. We evaluated our system on the dataset of (Mohammad et al., 2018), where it ranked 5th/1st rank in English/Spanish respectively.

2 Model

Our system is mainly composed of two parts: self-attention module and multiple independent CNNs as depicted in Figure 1. This structure is actually imitating how human perform the same task. In general, human firstly read a sentence and try to comprehend the meaning, which corresponds to self-attention in our system. Then human categorize the sentence to each emotion separately but not all at once, and that is the reason why our system use 11 independent CNNs. In addition to main structure, we added the description of preprocessing in the model description because it makes up a large proportion in NLP tasks, especially when

Proceedings of the 12th International Workshop on Semantic Evaluation (SemEval-2018), pages 141–145
New Orleans, Louisiana, June 5–6, 2018. ©2018 Association for Computational Linguistics

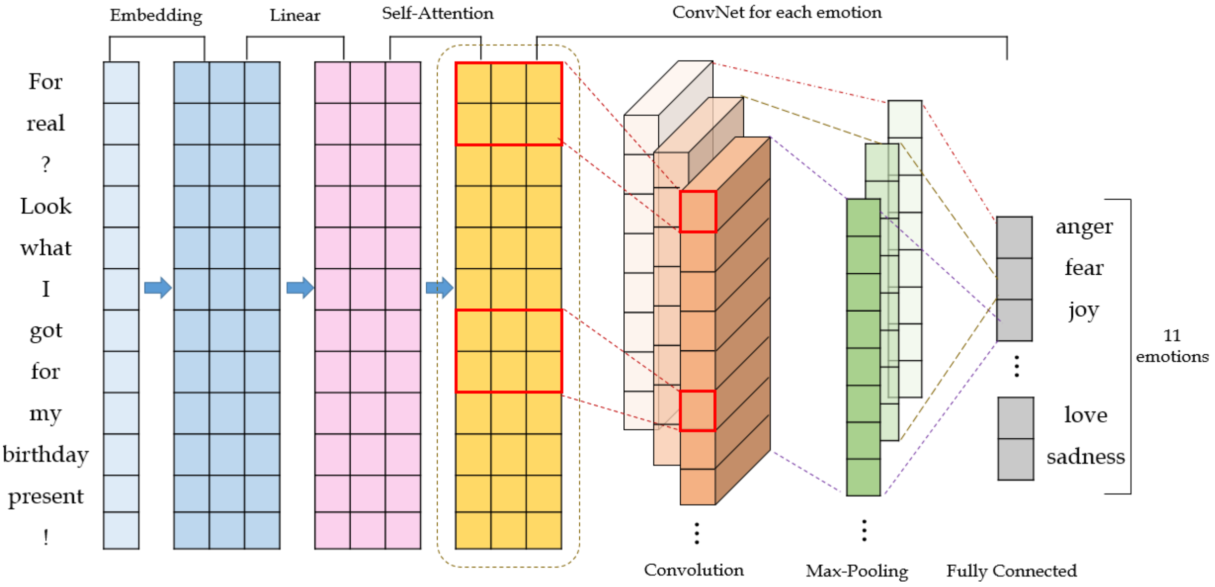

Figure 1: Overall architecture of the model. Preprocessed data goes through embedding layer, self-attention layer, Convolution layer and pooling layer step by step.

the dataset is small. Details are described in the following paragraph step by step.

Preprocessing: For raw data, we applied 3 steps of preprocessing:

(i) Our system mainly deals with limited numbers of tweet data, which is very noisy. In this case, preprocessing of data has crucial impact on model performance. Emoji may be referred to as a typical property of tweets and we found that considerable number of tweets contain emojis. Each emoji has a meaning of their own, and we converted every emoji in the data to phrase/word that represents its meaning. We call this procedure as **emoji-to-meaning** preprocessing. Some tweets have too many repetition of certain emoji that may make the sentence overbiased to certain emotions. Against expectations, removing overlapped emojis reduced performance.

(ii) Lower-case and tokenize data with **TweetTokenizer** in (Bird and Loper, 2002).

(iii) Remove all of the mentions and '#' symbols in the beginning of all topics. Unlike mentions, topics may include emotional words and hence we don't remove the topic itself.

Embedding: It is especially helpful to use pre-trained word embeddings when dealing with a small dataset. Among those well-known word embeddings such as Word2Vec(Mikolov et al., 2013), GloVe(Pennington et al., 2014) and fastText(Piotr et al., 2016), we adopt 300-dimension GloVe vectors for English ,which is trained on Common Crawl data of 840 billion tokens and 300-dimension fastText vectors for Spanish, which is trained on Wikipedia.

Self-attention: Vaswani et al. (2017) proposed a non-recurrent machine translation architecture called Transformer that is based on dot-product attention module. Usually, attention mechanisms are used as a submodule of deep learning models, calculating the importance weight of each position given a sequence. In our system, we adopt the self-attention mechanisms in (Vaswani et al., 2017) to represent sentences. The detailed structure of self-attention is shown in Figure 2. Dot-product of every embedded vector and weight matrix $W \in \mathbb{R}^{d_e \times 3d_e}$ is split through dimension as Q, K, V of the same size, where d_e is the dimensionality of embedded vectors. Then attended vector is computed as in (3).

$$E = [emb(x_1), emb(x_2), ..., emb(x_n)] \quad (1)$$

$$[Q, K, V] = [eW \ for \ e \ in \ E] \quad (2)$$

$$Attn(Q, K, V) = softmax(\frac{QK^T}{\sqrt{d_e}})V \quad (3)$$

Multi-head attention allows the model to benefit from ensemble effect only with the same amount

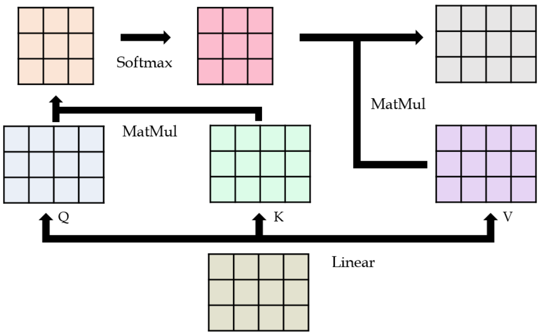

Figure 2: Inner architecture of self-attention module

of parameter.

$$Multihead(Q,K,V) = Concat(head_1, ..., head_h) \tag{4}$$

$$\text{where } head_i = Attn(Q_i, K_i, V_i)$$
$$Q = [Q_1, ..., Q_h],\ Q_i \in \mathbb{R}^{n \times \frac{d_e}{h}}$$
$$K = [K_1, ..., K_h],\ K_i \in \mathbb{R}^{n \times \frac{d_e}{h}}$$
$$V = [V_1, ..., V_h],\ V_i \in \mathbb{R}^{n \times \frac{d_e}{h}}$$

For each self-attention layer, there are additional position-wise feed-forward networks right after the attention submodule.

$$FFN(x) = max(0, xW_1 + b_1)W_2 + b_2 \tag{5}$$
$$\text{where } W_1 \in \mathbb{R}^{d_e \times d_f},\ W_2 \in \mathbb{R}^{d_f \times d_e} \tag{6}$$

In addition to these two sub-layers, there is a residual connection around each sub-layer, followed by layer normalization. Also, we can stack self-attention layers by substituting the embedding vectors in (1) with the output of the last self-attention layer.

Convolution & Pooling: Followed by self-attention layer are 11 independent 1-layer Convolution layers with max-pooling layers. Kim (2014) has proved that CNNs have lots of potential in sentence processing task and we adopt the CNNs in the same way.

Output & Loss: Each output of CNNs go through a fully-connected layer to generate a logit. Sigmoid activation is applied to calculate the probability of each emotion, and we use the sum of each class' cross-entropy as the final loss function.

3 Experiments & Results

3.1 Data

For the SemEval 2018 shared task, Mohammad et al.(2018) has provided tweet data with multiple labels among 11 pre-set emotions: 'angry', 'anticipation', 'disgust', 'fear', 'joy', 'love' 'optimism', 'pessimism', 'sadness', 'surprise' and 'trust'. We only use English and Spanish data among three different languages. The dataset consists of 6838/887/3259 tweets in English, 3561/679/2854 tweets in Spanish for train/validation/test data respectively.

3.2 Setup

We implemented a model with 3-layer self-attention and 1-layer CNN. With the restriction of fixed-size GloVe vector, we found that 300-dimension hidden state is excessive for such a small dataset that we added a position-wise linear layer between the embedding layer and self-attention layers to make $d_e = 30$. We employed $h = 2$ for multi-head attention and set $d_f = 64$. Two regularization techniques are applied to our system: Dropout with $P_{drop} = 0.1$ for self-attention, and L2 regularization for all weight matrix but not bias. We added 0.001 times regularization loss to original loss function. We optimized the loss with Gradient Descent using Adam optimization algorithm with additional learning rate decay.

3.3 Model variants

We conduct experiments with following variants of our model.

- **AC**: Self-attention + CNNs, which is our basic system.
- **AC - attn**: Basic system without self-attention module.
- **AC + nrc1**: We mainly used NRC Emotion lexicon(Mohammad and Turney, 2013) to make word-level label of each sentence, counting the occurence of each emotion in the sentence. Each of the word-level label is concatenated to the output vector of each pooling layer.
- **AC + nrc2**: At evaluation/test step, binarize the word-level label and add 0.4 times the label value to the logit.
- **AC + synth**: Inspired by (Sennrich et al., 2016), we made synthetic data using unlabeled SemEval-2018 AIT DISC data[1] with

[1] https://www.dropbox.com/s/2phcvj300lcdnpl/SemEval2018-AIT-DISC.zip?dl=0

pre-trained model, and fine-tuned the model with synthetic data.

3.4 Experimental results

We conduct several experiments to prove the effectiveness of our model, each to verify the benefit from: (1) tweets specific preprocessing (2) self-attention representation (3) emotional lexicon utilization. Experimental results are **mainly compared with English data.**

3.4.1 Impact of emoji-to-meaning

We firstly verify the efficiency of emoji-to-meaning preprocessing. Table 1 shows the accuracies of the same model with different preprocessing. We found that emoji-to-meaning preprocessing can improve the model accuracy by 1%. When a emoji is converted to its meaning, it can be represented as a combination of emotional words allowing it to not only reduce redundant vocabulary but also further emphasize the influence of certain emotions.

Model	Accuracy(valid)	Accuracy(test)
AC (w/o)	54.86%	54.91%
AC	55.94%	55.90%

Table 1: Experimental results with and without emoji-to-meaning preprocessing.

3.4.2 Impact of self-attention

To examine the effectiveness of self-attention representation, we simply get rid of self-attention layers. Table 2 shows that by removing the self-attention layers, both the validation/test accuracy dropped over 4%. This may be attributed to the ability of self-attention: It helps the model to better learn the long-range dependency of sentences. Learning long-range dependencies is a key challenge in NLP tasks and self-attention module can shorten the length of paths forward and backward signals have to traverse in the network as described in (Vaswani et al., 2017).

Model	Accuracy(valid)	Accuracy(test)
AC - attn	51.04%	51.60%
AC	55.94%	55.90%

Table 2: Comparison between our basic system and basic system without self-attention module.

3.4.3 Impact of extra resources

Lack of data has crucial impact on model generalization. Generalization techniques such as dropout or L2 regularization can relieve over-fitting problem to a certain extent; however, it can't totally substitute the effect of rich data. So we apply some heuristic methods to exploit extra resources as described in 3.3. Table 2 shows that model can slightly benefit from extra lexicon if used properly. However, adding synthetic data which is made from pre-trained model didn't help a lot, and in some cases even reduce the accuracy of the test result. Actually, Sennrich et al.(2016) emphasized that they used the monolingual sentences as the target sentences, informing that the target-side information, which corresponds to label in our task, is not synthetic. However, we made synthetic labels with a pre-trained model and it may only cause over-fitting problem to the original training data.

Model	Accuracy(valid)	Accuracy(test)
AC	55.94%	55.90%
AC + nrc1	56.13%	56.02%
AC + nrc2	57.16%	56.40%
AC + synth	55.88%	55.90%
Ensemble	**59.76%**	**57.40%**

Table 3: Experimental results with extra resources and an ensemble result

3.4.4 Ensemble

Our best results are obtained with an ensemble of 9 parameter sets of AC + nrc2 model that differ in their random initializations. The ensemble model achieved validation/test accuracy of 59.76%/57.40% in English data and 50.00%/46.90% in Spanish data respectively.

4 Conclusion

In this paper, we proposed an attention-based sentence classifier that can classify a sentence into multiple emotions. Experimental results demonstrated that our system has effective structure for sentence understanding. Our system shallowly follows human's procedure of classifying sentences into multiple labels. However, some emotions may have some relatedness while our model treats them independently. In our future work, we would like to further take those latent relation among emotions into account.

Acknowledgments

This work was supported by the National Research Foundation of Korea(NRF) funded by the Korea government(MSIT) (No. 2016M3C4A7952632), Industrial Strategic Technology Development Program(No. 10073144) funded by the Ministry of Trade, Industry & Energy(MOTIE, Korea)

References

Dzmitry Bahdanau, Kyunghyun Cho, and Yoshua Bengio. 2015. Neural machine translation by jointly learning to align and translate. *International Conference on Learning Representations Workshop*.

Steven Bird and Edward Loper. 2002. Nltk: the natural language toolkit. In *Proceedings of the ACL-02 Workshop on Effective tools and methodologies for teaching natural language processing and computational linguistics*.

Misha Denil, Loris Bazzani, Hugo Larochelle, and Nando de Freitas. 2012. Learning where to attend with deep architectures for image tracking. *Neural Computation*.

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

Yoon Kim. 2014. Convolutional neural networks for sentence classification. In *Proceedings of the 2014 Conference on Empirical Methods in Natural Language Processing*, pages 1746–1751. Association for Computational Linguistics.

Tomas Mikolov, Ilya Sutskever, Kai Chen, Greg Corrado, and Jeff Dean. 2013. Distributed representations of words and phrases and their compositionality. *Advances in Neural Information Processing Systems*, 26:3111–3119.

Saif Mohammad and Peter Turney. 2013. Crowdsourcing a word-emotion association lexicon. *Computational Intelligence*, 29(3):436–465.

Saif M. Mohammad, Felipe Bravo-Marquez, Mohammad Salameh, and Svetlana Kiritchenko. 2018. Semeval-2018 Task 1: Affect in tweets. In *Proceedings of International Workshop on Semantic Evaluation (SemEval-2018)*, New Orleans, LA, USA.

Jeffrey Pennington, Richard Socher, and Christopher D.Manning. 2014. Glove: Global vectors for word representation. In *Proceedings of the Conference on Empirical Methods in Natural Language Processing*.

Bojanowski Piotr, Grave Edouard, Joulin Armand, and Mikolov Tomas. 2016. Enriching word vectors with subword information. *arXiv preprint arXiv:1607.04606*.

Rico Sennrich, Barry Haddow, and Alexandra Birch. 2016. Improving neural machine translation models with monolingual data. In *Proceedings of the 54th Annual Meeting of the Association for Computational Linguistics*, volume 1, pages 86–96.

Martin Sundermeyer, Ralf Schluter, and Hermann Ney. 2010. Lstm neural networks for language modeling. *Interspeech*, pages 194–197.

Ashish Vaswani, Noam Shazeer, Niki Parmar, Jakob Uszkoreit, Llion Jones, Aidan N. Gomez, Lukasz Kaiser, and Illia Polosukhin. 2017. Attention is all you need. *Annual Conference on Neural Information Processing Systems*.