Improving Multi-label Emotion Classification by Integrating both General and Domain Knowledge

Wenhao Ying **Rong Xiang** **Qin Lu**
The Hong Kong Polytechnic University, Hong Kong, China
yingwh123@sina.com, {csrxiang,csluqin}@comp.polyu.edu.hk

Abstract

Deep learning based general language models have achieved state-of-the-art results in many popular tasks such as sentiment analysis and QA tasks. Text in domains like social media has its own salient characteristics. Domain knowledge should be helpful in domain relevant tasks. In this work, we devise a simple method to obtain domain knowledge and further propose a method to integrate domain knowledge with general knowledge based on deep language models to improve performance of emotion classification. Experiments on Twitter data show that even though a deep language model fine-tuned by a target domain data has attained comparable results to that of previous state-of-the-art models, this fine-tuned model can still benefit from our extracted domain knowledge to obtain more improvement. This highlights the importance of making use of domain knowledge in domain-specific applications.

1 Introduction

Deep language models (LM) have been very successful in recent years. In pre-training, a deep LM learns to predict unseen words in the context at hand in an unsupervised way, which enables the LM to make use of very large amount of unlabeled data. By using deep structures and large amount of training data, these deep LMs can learn useful linguistic knowledge common to many natural language processing tasks. For example, BERT (Devlin et al., 2019) has the ability to encode grammatical knowledge in context in its representations (Hewitt and Manning, 2019). Deep LMs provide general knowledge of text to benefit downstream tasks. To be adaptive to a target domain, they do need to be fine-tuned by data of the target domain.

Obviously, every domain has its own characteristics which deserve special attention. A typical example is Twitter data. In twitter, people can express their thoughts online in real time. Due to its informal nature, people tend to pick whatever comes to their mind to jot down their opinions even if the writing does not conform to grammar rules. For example, Combinations of characters, such as ":(" and ":-)", are often used to express different emotions. Deliberate irregular spellings also occur in Twitter to indicate authors' attitude. Table 1 shows an example of an irregular expression and how it can be preprocessed at word level, wordpiece level and at domain level. Many of these domain-specific expressions are strong indicators for affective analysis in Twitter, and these characteristics are worthy of special consideration. Simply neglecting them would lose a lot of useful information. Such information can be formulated as domain knowledge by using Twitter preprocessor like (Baziotis et al., 2017). After Twitter-specific preprocessing, these expressions are annotated automatically and we can find informa-

Snippet	haaapppyyyy birthday best friend!! Love you lots #love
Word	'haaapppyyyy', 'birthday', 'best', 'friend', '!', '!', 'Love', 'you', 'lots', '#', 'love'
Wordpiece	'ha', '##aa', '##pp', '##py', '##y', '##y', '##y', 'birthday', 'best', 'friend', '!', '!', 'Love', 'you', 'lots', '#', 'love'
Domain-specific	'happy', '<elongated>', 'birthday', 'best', 'friend', '!', '<repeated>', 'Love', 'you', 'lots', '</hashtag>', 'love', '<hashtag>'

Table 1: Example of an real-world irregular expression preprocessed by methods at different levels. '##' is a sign of word pieces, and '<>' is a special mark produced by a Twitter-specific preprocessor.

Proceedings of the 2019 EMNLP Workshop W-NUT: The 5th Workshop on Noisy User-generated Text, pages 316–321
Hong Kong, Nov 4, 2019.

tive token patterns from preprocessed tweets. In the above example, a pattern '[+, <elongated>]' expresses more positive sentiment than a regular positive word. Another pattern '[</hashtag>, *, </hashtag>]' means it is a hashtag and usually has an overall meaning for a tweet.

In this work, we select the popular BERT language model to provide general linguistic knowledge for modelling sentences. As a commonly used deep LM, BERT is not intended to pay attention to domain-specific details in Twitter. BERT actually use sub-word tokens as its inputs for generalization, and a word is first divided into a number of smaller units if necessary before being converted to embeddings. We design a token pattern detector that sifts through preprocessed tweets to obtain domain knowledge, and supplement BERT with extracted domain-specific features. To integrate the domain knowledge with BERT, we first fine-tune BERT to extract general features of Twitter data. Features from the fine-tuned BERT are then integrated with domain-specific features to classify tweets into target emotions. Performance evaluations show that even though BERT was pretrained on different source domains, the fine-tuned BERT using Twitter data indeed attains comparable results to that of the previous state-of-the-art models. Most importantly, even after BERT is tuned by Twitter data, integration of domain knowledge in our system still makes over one percent improvement on the accuracy of emotion classification compared to the previous state-of-the-art method using BERT only.

2 Related Work

Related works include both deep LMs especially BERT, a representative deep learning based LM and works on Twitter classification.

2.1 Deep Language Models

In contrast to n-gram LMs and early neural models for learning word embeddings, recent LMs have deeper structures. ELMo (Peters et al., 2018) use a stack of bi-directional LSTM to encode word context either from left-to-right or from right-to-left. BERT (Devlin et al., 2019) has a bidirectional structure to learn context from both directions. As a consequence of its bidirectionality, BERT is not trained by predicting words in sequence either from left-to-right or from right-to-left. After masking a part of words in a sentence, training predicts the masked and unseen words within the remaining context. However, by corrupting inputs with masks, BERT neglects dependency between masked positions. XLNet (Yang et al., 2019) proposes to maximizes the likelihood over all permutations of the factorization order of conditional probability to learn bidirectional context without masking. Recently, RoBERTa (Liu et al., 2019) matches the previous state-of-the-art language models by training BERT on even larger data with optimized hyper-parameters.

In this work, we use BERT as our baseline, a popular deep language model. BERT has a stack of transformer layers (Vaswani et al., 2017). The central part of a transformer is a multi-head attention mechanism to include queries, keys, and values as inputs, which makes scaled dot-product attention among all inputs. Let Q denote a query matrix, K denote a key matrix, V denote a value matrix, and $Q = K = V$ in the case of BERT. The scaled dot-product attention formula is then given as follows:

$$\text{Attention}(Q, K, V) = \text{softmax}(\frac{QK^T}{\sqrt{d_k}})V$$

where d_k is the dimension of queries and keys. For BERT, an input token has a positional embedding and a segment embedding in addition to its regular word embedding. Positional embeddings tell BERT relative positions of two words and segment embeddings help BERT to differentiate two sentences of a pair. In each sentence fed into BERT, a special token [CLS] is inserted at the first place and one uses its corresponding output as the overall representation of this sentence for sentence-level tasks such as entailment or sentiment analysis.

2.2 Twitter Affective Analysis

As a platform to express everyday thoughts, Twitter has huge amount of affect-related text. Thus Twitter is a good source of research study on affective analysis of people towards a topic. N-grams and negative indicators are widely used in affective analysis of Twitter (Mohammad et al., 2013; Miura et al., 2014). Affect-based lexicons are also included to provide general sentiment or emotion information (Hagen et al., 2015). (Go et al.) use :) and :(emoticons as natural labels and collect a pseudo-labeled training data to increase their n-gram classifier. Similarly, Wang et al (Wang et al.,

2012) look for tweets with a target set of hashtags such as #happy and #sad to collect an emotion-linked training data. Due to the abundance of these naturally labeled training data, deep neural networks has proven its dominance in recent competitions by means of the framework of transfer learning (Severyn and Moschitti, 2015; Deriu et al., 2016; Cliche, 2017). They pre-train models on naturally labeled data to get a better starting point and fine-tune their models on the target task.

3 Methodology

The basic idea of our work is to use a Twitter-specific preprocessor to decode Twitter-related expressions. A token pattern detector is then trained to identify affect-bearing token patterns. Finally, a two-step training process is introduced to integrate general knowledge and the detected domain knowledge for emotion classification.

3.1 Domain Specific Information Extraction

Because tweets are informal text with a lot of expression variations, we first use the Twitter preprocessing tool *ekphrasis* (Baziotis et al., 2017) to obtain domain-related information. *ekphrasis* conducts Twitter-specific tokenization, spell checking correction, text normalization and word segmentation. It recognizes many special expressions like emoticons, dates and times with an extensive list of regular expressions. Tokens can also be split further to obtain useful information. A typical example is to split hashtags. After tokenization, expressions with a lot of variations such as user handles and URLs are normalized with designated marks. The result can properly align tokens to their regular forms in the vocabulary without loss of information nor the need to enlarge vocabulary size. Table 2 give a few examples of preprocessed words with annotation, where <*>is a designated annotation mark.

Original	Processed
':)', ':-)'	<happy>
'REAL'	<allcaps>real </allcaps>
'goooooood'	good <elongated>
October 8th	<date>
@jeremy	<user>
#Christmas	<hashtag>Christmas </hashtag>

Table 2: Examples of typical Twitter-specific expressions and their preprocessed versions with annotation marks

3.2 Token Pattern Detector

After Twitter-specific annotation using a preprocessing tool, some input words are annotated and stand out conspicuously. In this step, we identify informative token patterns for emotion classification. A simple convolution network is used to examine tokens within a fixed-length window to detect token patterns. The network structure is a 1D convolution layer followed by temporal max-pooling, similar to that of (Kim, 2014). But we only use a token window of size 3 to simply observe trigrams. The three-token range should cover most of potential token patterns for our work. Given a convolution kernel, it serves as a detector to check whether a particular token pattern appears in a sentence measured by a matching score s_i according to the following formula 1,

$$s_i = w^T[e_i, e_{i+1}, e_{i+2}] + b \quad (1)$$

where e_i, e_{i+1} and e_{i+2} are word embeddings corresponding to successive tokens at positions $i, i+1, i+2$, w and b are learnable parameters of this kernel. A detector moves through all possible subsequences and produces a list $\{s_1, s_2, \cdots, s_{n-2}\}$. The following temporal max-pooling obtains the maximum value from the list as an indicator suggesting whether a sentence includes a particular token pattern. Hundreds of such detectors are used together to find various types of token patterns. All the outputs of max-pooling for each detector make up the domain-specific representation for a sentence.

3.3 Multi-label Emotion Classification

A two-step training process is designed to integrate general and domain knowledge in multi-label emotion classification. In the first step, we fine-tune BERT on the training data of our target task initialized with pre-trained parameters[1]. The model follows the same input format as pre-training in which a word is divided into several word pieces before they are fed into BERT. Then, we use the output for [CLS] from the last layer as the general feature representation of a sentence. We also train a convolutional detector from scratch on the training data with Twitter-specific annotation and use the output from the last layer as a sentence's domain-specific features. The parameters

[1]Pre-trained BERT models are obtained from https://github.com/huggingface/pytorch-transformers

of both models are fixed after this step and therefore the representation produced by each model will not be changed in the next step. In the second step, the two types of representations are concatenated and fed into a linear scoring layer for emotion class predication. For a target emotion i and the representation of a sentence x, its score is computed by $\hat{y}[i] = w^T x$. The layer is tuned on the training data so that general and Twitter-specific features can work collaboratively.

For the gold labels y and prediction scores $\hat{y}$, their loss is given by

$$loss(x, y) = -\frac{1}{C}\sum_{i}^{C}(y[i] * \log(\frac{1}{1+e^{-\hat{y}[i]}}) + (1-y[i]) * \log(\frac{e^{-\hat{y}[i]}}{1+e^{-\hat{y}[i]}})) \quad (2)$$

where $\hat{y}[i]$ and $y[i]$ are for the i_{th} emotion class, and C is the number of target emotion classes. If the target emotion class is positive, that is $y[i] = 1$, the loss function requires the corresponding prediction to be as large as possible. When making prediction of a target emotion for a sample, we assign it a positive label if $\hat{y}[i] \geq 0$.

4 Performance Evaluation

Performance evaluation is conducted on multi-label emotion classification of SemEval-2018 Task 1 (Mohammad et al., 2018). Given a tweet, the task requires participants to classify text to zero or more of 11 target emotions.

4.1 Setup

SemEval-2018 dataset was already split into training, development and testing sets by its organizer. We train and tune our models on the training and development sets, and report classification results on the testing set. Word embeddings of our CNN detector are learned from a corpus of 550M unlabeled tweets by word2vec (Mikolov et al., 2013) [2]. Multi-label accuracy, known as Jaccard Index, is used as the evaluation metric, defined as the size of the intersection divided by the size of the union of the true label set and predicted label set. Macro-F1 and Micro-F1 are used as secondary evaluation metrics following the same practice of SemEval-2018 Task 1. In the two-step training, we first train our CNN detector and fine-tune BERT on the training data 10 times and select the parameters with the best performance on the development set to hopefully provide good representation of both general and domain-specific information. In the second step, the representation for a tweet remains unchanged and only the parameters of a scoring layer is learned.

[2] We use the pre-trained embeddings from (Baziotis et al., 2018)

4.2 Evaluation

Table 3 lists the results of multi-label emotion classification on SemEval-2018. The first blocks are the state-of-the-art models on SemEval-2018 Task 1, where we directly cite the results from their papers. Two BERT models are used as additional baselines including $\text{BERT}_{\text{base}}$, which has 12 layers of transformers with 768 dimension, and $\text{BERT}_{\text{large}}$, which has 24 layers of transformers with 1024 dimension. BERT using domain knowledge (DK) proposed by our work are appended with '+DK'. Another baselines include a biLSTM and our CNN detector. To randomize parameter initialization and learning algorithm, we train CNN and BiLSTM from scratch, fine-tune BERT from the given initialized parameters, and learn the weights of scoring layer 10 times, respectively. We report the average performance on the testing set for each model.

Model	acc.	micro F1	macro F1
PlusEmo2Vec (Park et al., 2018)	57.6	69.2	49.7
TCS Research (Meisheri and Dey, 2018)	58.2	69.3	53.0
NTUA-SLA (Baziotis et al., 2018)	58.8	70.1	52.8
CNN Detector	55.8	68.5	50.0
BiLSTM	56.3	68.7	51.0
$\text{BERT}_{\text{base}}$	58.4	70.4	54.2
$\text{BERT}_{\text{large}}$	58.8	70.7	55.2
$\text{BERT}_{\text{base}}$+DK	59.1†‡	71.3†‡	54.9†
$\text{BERT}_{\text{large}}$+DK	**59.5**†‡	**71.6**†‡	**56.3**†

Table 3: Result of multi-label emotion classification on SemEval-2018. † means the result is statistically significant with $p < 0.01$ in contrast to the state-of-the-art NTUA-SLA. ‡ means the improvement by integrating domain knowledge is statistically significant with $p < 0.01$ compared with its corresponding pure BERT, as $\text{BERT}_{\text{base}}$ vs $\text{BERT}_{\text{base}}$+DK. For Macro-F1, the results are statistically significant with $p < 0.05$.

As expected, the CNN detector has the worst

Emotion	$\text{BERT}_{\text{large}}$	+DK	
anger	78.82	79.34	(+0.52)
anticipation	23.60	23.60	(+0.00)
disgust	75.00	76.28	(+1.28)
fear	74.96	76.18	(+1.22)
joy	85.22	86.39	(+1.17)
love	61.52	63.94	(+2.42)
optimism	73.41	73.73	(+0.32)
pessimism	32.05	32.16	(+0.11)
sadness	70.21	71.90	(+1.69)
surprise	22.56	26.24	(+3.68)
trust	9.56	9.03	(-0.53)

Table 4: F1 on binary classification for each emotion class.

performance as this tri-gram model is too simple to learn complex relationships such as long-distance negative relations. The CNN detector is to sift through domain-specific token patterns to supplement the general knowledge of BERT. Both of the fine-tuned pure BERTs are either comparable or slightly better than the performance of the previous state-of-the-art models. With the abundant pre-training data and their deep structure, BERT models obtain a good starting point for a domain-specific task. More importantly, both BERT models benefit from domain knowledge supplied by the CNN detector to obtain performance improvement of 1.20% on major multi-label accuary for both models. Both BERT models with domain knowledge outperform their corresponding pure BERT and the state-of-the-art model statistically significantly with $p < 0.01$ except for Macro-F1 where the results are statistically significant with $p < 0.05$. In the first-step training, the selected CNN, $\text{BERT}_{\text{base}}$ and $\text{BERT}_{\text{large}}$ for providing tweet representation has accuracy of 56.7%, 59.0% and 58.9%, respectively. Table 3 shows that BERT integrating with Twitter-specific features outperforms both general and domain-specific component model.

For more detailed investigation of the effect of domain knowledge, Table 4 shows the result of binary classification for each emotion class measured by F1 score. Improvements are obtained in nine out of eleven emotion classes. If excluding 'surprise' and 'trust' which have low percentage of occurrence, salient improvements come mostly from 'disgust', 'fear', 'joy', 'love' and 'sadness'. Abundant domain-specific expressions in Twitter, such as emoticons ':-)' and ':-(' and hashtags like '#offended', are useful affective indicators, which are not used fully by BERT.

5 Conclusion

In this work, we leverage deep language models to provide general sentence representations and integrate them with domain knowledge. We show that integration of both types of knowledge improves multi-label emotion classification of tweets. Evaluation shows that a deep LM like BERT has the capacity to perform well. Yet, its performance can still be further improved by integrating elaborate domain knowledge. Future works may investigate other deep LMs as well as data of other domains.

Acknowledgments

This work was partially supported by a GRF grant from HKUGC (PolyU 152006/16E).

References

Christos Baziotis, Athanasiou Nikolaos, Alexandra Chronopoulou, Athanasia Kolovou, Georgios Paraskevopoulos, Nikolaos Ellinas, Shrikanth Narayanan, and Alexandros Potamianos. 2018. Ntua-slp at semeval-2018 task 1: Predicting affective content in tweets with deep attentive rnns and transfer learning. In *Proceedings of The 12th International Workshop on Semantic Evaluation*, pages 245–255.

Christos Baziotis, Nikos Pelekis, and Christos Doulkeridis. 2017. Datastories at semeval-2017 task 4: Deep lstm with attention for message-level and topic-based sentiment analysis. In *Proceedings of the 11th International Workshop on Semantic Evaluation (SemEval-2017)*, pages 747–754, Vancouver, Canada. Association for Computational Linguistics.

Mathieu Cliche. 2017. Bb_twtr at semeval-2017 task 4: Twitter sentiment analysis with cnns and lstms. In *Proceedings of the 11th International Workshop on Semantic Evaluation (SemEval-2017)*, pages 573–580.

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*, CONF, pages 1124–1128.

Jacob Devlin, Ming-Wei Chang, Kenton Lee, and Kristina Toutanova. 2019. 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.

Alec Go, Richa Bhayani, and Lei Huang. Twitter sentiment classification using distant supervision.

Matthias Hagen, Martin Potthast, Michel Büchner, and Benno Stein. 2015. Webis: An ensemble for twitter sentiment detection. In *Proceedings of the 9th international workshop on semantic evaluation (SemEval 2015)*, pages 582–589.

John Hewitt and Christopher D Manning. 2019. A structural probe for finding syntax in word representations. 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 4129–4138.

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

Yinhan Liu, Myle Ott, Naman Goyal, Jingfei Du, Mandar Joshi, Danqi Chen, Omer Levy, Mike Lewis, Luke Zettlemoyer, and Veselin Stoyanov. 2019. Roberta: A robustly optimized bert pretraining approach. *arXiv preprint arXiv:1907.11692*.

Hardik Meisheri and Lipika Dey. 2018. Tcs research at semeval-2018 task 1: Learning robust representations using multi-attention architecture. In *Proceedings of The 12th International Workshop on Semantic Evaluation*, pages 291–299.

Tomas Mikolov, Ilya Sutskever, Kai Chen, Greg S Corrado, and Jeff Dean. 2013. Distributed representations of words and phrases and their compositionality. In *Advances in neural information processing systems*, pages 3111–3119.

Yasuhide Miura, Shigeyuki Sakaki, Keigo Hattori, and Tomoko Ohkuma. 2014. Teamx: A sentiment analyzer with enhanced lexicon mapping and weighting scheme for unbalanced data. In *Proceedings of the 8th International Workshop on Semantic Evaluation (SemEval 2014)*, pages 628–632.

Saif Mohammad, Svetlana Kiritchenko, and Xiaodan Zhu. 2013. Nrc-canada: Building the state-of-the-art in sentiment analysis of tweets. In *Second Joint Conference on Lexical and Computational Semantics (* SEM), Volume 2: Proceedings of the Seventh International Workshop on Semantic Evaluation (SemEval 2013)*, pages 321–327.

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.

Ji Ho Park, Peng Xu, and Pascale Fung. 2018. Plusemo2vec at semeval-2018 task 1: Exploiting emotion knowledge from emoji and# hashtags. *arXiv preprint arXiv:1804.08280*.

Matthew Peters, Mark Neumann, Mohit Iyyer, Matt Gardner, Christopher Clark, Kenton Lee, and Luke Zettlemoyer. 2018. Deep contextualized word representations. In *Proceedings of the 2018 Conference of the North American Chapter of the Association for Computational Linguistics: Human Language Technologies, Volume 1 (Long Papers)*, pages 2227–2237.

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)*, pages 464–469.

Ashish Vaswani, Noam Shazeer, Niki Parmar, Jakob Uszkoreit, Llion Jones, Aidan N Gomez, Łukasz Kaiser, and Illia Polosukhin. 2017. Attention is all you need. In *Advances in neural information processing systems*, pages 5998–6008.

Wenbo Wang, Lu Chen, Krishnaprasad Thirunarayan, and Amit P Sheth. 2012. Harnessing twitter" big data" for automatic emotion identification. In *2012 International Conference on Privacy, Security, Risk and Trust and 2012 International Confernece on Social Computing*, pages 587–592. IEEE.

Zhilin Yang, Zihang Dai, Yiming Yang, Jaime Carbonell, Ruslan Salakhutdinov, and Quoc V Le. 2019. Xlnet: Generalized autoregressive pretraining for language understanding. *arXiv preprint arXiv:1906.08237*.