

CAiRE_HKUST at SemEval-2019 Task 3: Hierarchical Attention for Dialogue Emotion Classification

Genta Indra Winata*, Andrea Madotto*, Zhaojiang Lin,
Jamin Shin, Yan Xu, Peng Xu, Pascale Fung

Center for Artificial Intelligence Research (CAiRE)

Department of Electronic and Computer Engineering

The Hong Kong University of Science and Technology, Clear Water Bay, Hong Kong

{giwinata, amadotto, zlinao}@connect.ust.hk,

{jmshinaa, yxucb, pxuab}@connect.ust.hk, pascale@ece.ust.hk

Abstract

Detecting emotion from dialogue is a challenge that has not yet been extensively surveyed. One could consider the emotion of each dialogue turn to be independent, but in this paper, we introduce a hierarchical approach to classify emotion, hypothesizing that the current emotional state depends on previous latent emotions. We benchmark several feature-based classifiers using pre-trained word and emotion embeddings, state-of-the-art end-to-end neural network models, and Gaussian processes for automatic hyper-parameter search. In our experiments, hierarchical architectures consistently give significant improvements, and our best model achieves a 76.77% F1-score on the test set.

1 Introduction

Customer service can be challenging for both the givers and receivers of services, leading to emotions on both sides. Even human service-people who are trained to deal with such situations struggle to do so, partly because of their own emotions. Neither do automated systems succeed in such scenarios. *What if we could teach machines how to react under these emotionally stressful situations of dealing with angry customers?*

This paper represents work on the SemEval 2019 shared task (Chatterjee et al., 2019b), which aims to bring more research on teaching machines to be empathetic, specifically by contextual emotion detection in text. Given a textual dialogue with two turns of context, the system has to classify the emotion of the next utterance into one of the following emotion classes: Happy, Sad, Angry, or Others. The training dataset contains

15K records for emotion classes, and contains 15K records not belonging to any of the aforementioned emotion classes.

The most naive first step would be to recognize emotion from a given flattened sequence, which has been researched extensively despite the very abstract nature of emotion (Socher et al., 2013; Felbo et al., 2017a; McCann et al., 2017; Xu et al., 2018; Chatterjee et al., 2019a). However, these *flat* models do not work very well on dialogue data as we have to merely concatenate the turns and flatten the hierarchical information. Not only does the sequence get too long, but the hierarchy between sentences will also be destroyed (Hsu and Ku, 2018; Kim et al., 2018). We believe that the natural flow of emotion exists in dialogue, and using such hierarchical information will allow us to predict the last utterance’s emotion better.

Naturally, the next step is to be able to detect emotion with a hierarchical structure. To the best of our knowledge, this task of extracting emotional knowledge in a hierarchical setting has not yet been extensively explored in the literature. Therefore, in this paper, we investigate this problem in depth with several strong hierarchical baselines and by using a large variety of pre-trained word embeddings.

2 Methodology

In this task, we focus on two main approaches: 1) feature-based and 2) end-to-end. The former compares several well-known pre-trained embeddings, including *GloVe* (Pennington et al., 2014), *ELMo* (Peters et al., 2018), and *BERT* (Devlin et al., 2018), as well as emotional embeddings. We combine these pre-trained features with a simple Logistic Regression (LR) and XGBoost (Chen and Guestrin, 2016) model as the classifier to compare their effectiveness. The latter approach is to train

*Equal contribution.

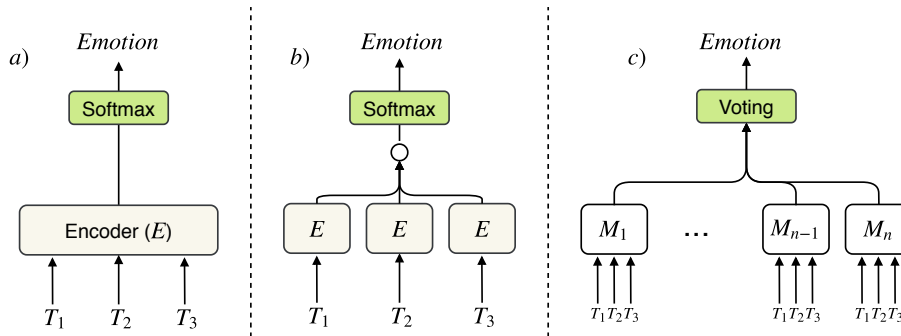


Figure 1: a) Flat model; b) Hierarchical model; c) Voting scheme

a model fully end-to-end with back-propagation. We mainly compare the performances of *flat* models and *hierarchical* models, which also take into account the sequential turn information of dialogues.

2.1 Feature-based Approach

The pre-trained feature-based approach can be subdivided into two categories: 1) word embeddings pre-trained only on semantic information, and 2) emotional embeddings that augment word embeddings with emotional or emoji information. We also examine the use of both categories.

Word Embeddings These include the standard pre-trained non-contextualized GloVe (Pennington et al., 2014), the contextualized embeddings from the bidirectional long short term memory (biLSTM) language model ELMo (Peters et al., 2018), and the more recent *transformer* based embeddings from the bidirectional language model BERT (Devlin et al., 2018).

Emotional Embeddings These refer to two types of features equipped with emotional knowledge. The first is a word-level emotional representation called Emo2Vec (Xu et al., 2018). It is trained with six different emotion-related tasks and has shown extraordinary performance over 18 different datasets. The second is a sentence-level emotional representation called DeepMoji (Felbo et al., 2017b), trained with a biLSTM with an attention model to predict emojis from text on a 1,246 million tweet corpus. Finally, we use Emoji2Vec (Eisner et al., 2016) which directly maps emojis to continuous representations.

2.2 End-to-End Approach

We consider four main models for the end-to-end approach: fine-tuning ELMo (Peters et al.,

2018), fine-tuning BERT (Devlin et al., 2018), Long Short Term Memory (LSTM) (Hochreiter and Schmidhuber, 1997), and Universal Transformer (UTRS) (Dehghani et al., 2018).¹ In the latter model, we also run a Gaussian process for automatic hyper-parameter selection.

ELMo This model from Peters et al. (2018) is a deep contextualized embedding extracted from a pre-trained bidirectional language model that has shown state-of-the-art performance in several natural language processing (NLP) tasks.

BERT This is the state-of-the-art bidirectional pre-trained language model that has recently shown excellent performance in a wide range of NLP tasks. Here, we use $BERT_{BASE}$ ² as our sentence encoder. However, the original model failed to capture the emoji features due to the fact that all the emoji tokens are missing in the vocab. Therefore, we concatenate each sentence representation from BERT with bag of words *Emoji2Vec* (Eisner et al., 2016). Then, a UTRS is used as a context encoder to encode the whole sequence.

LSTM and Universal Transformer LSTM is the widely known model used almost ubiquitously in the literature, while UTRS is a recently published recurrent extension of the multi-head self-attention based model, Transformer from (Vaswani et al., 2017). Finally, for all models, we consider a hierarchical extension which considers the turn information as well. We add another instance of the same model to also encode sentence-level information on top of the word-level representations. We also apply word-level attention to

¹We also tested Transformer, but had an overfitting issue

²We used a PyTorch implementation from <https://github.com/huggingface/pytorch-pretrained-BERT>

Table 1: The table shows the F1 score on LR and XGBoost.

Feature(s)	Classifier	F1
DeepMoji	LR	64.87
ELMo	LR	63.86
GLoVe	LR	55.11
Emo2Vec	LR	50.91
BERT	LR	44.51
Emoji2Vec	LR	30.45
ELMo + DeepMoji	LR	65.63
ELMo + Emo2Vec	LR	65.42
Emoji2Vec + GLoVe	LR	58.00
ELMo + DeepMoji	XGBoost	69.86

Table 2: The table shows F1 score on flat and hierarchical end-to-end models. GP denotes as Gaussian process.

Model	Flat	Hierarchical
LSTM	72.53	73.45
LSTM+GLoVe	73.95	75.64
LSTM+GLoVe+Emo2Vec	73.85	74.59
UTRS	72.41	74.06
ELMo	68.14	70.55
BERT	66.12	73.29

Table 3: The table shows F1 score on different ensemble models. XGB denotes XGBoost with ELMo and DeepMoji features. ALL denotes all ensemble models.

Model	F1
Ensemble ₁ (3 HLSTMs)	76.08
Ensemble ₂ (HBERT + HLSTM + HUTRS)	75.76
Ensemble ₃ (HBERT + 3 HLSTMs + HUTRS)	76.26
Ensemble ₄ (HBERT + 5 HLSTMs + HUTRS)	76.24
Ensemble ₅ (HBERT + 5 HLSTMs + HUTRS)	76.20
Ensemble_{final} (ALL + HLSTM + XGB)	76.77
- Angry	75.88
- Happy	73.65
- Sad	81.30

select the important information words on each dialogue turn.

3 Evaluation

In this section, we present the evaluation metrics used in the experiment, followed by results on feature-based, end-to-end, and ensemble approaches and Gaussian process search.

3.1 Training Details

Feature-Based For the feature-based approach, we run LR and XGBoost on features using the Scikit-Learn toolkit (Pedregosa et al., 2011) without any additional tuning.

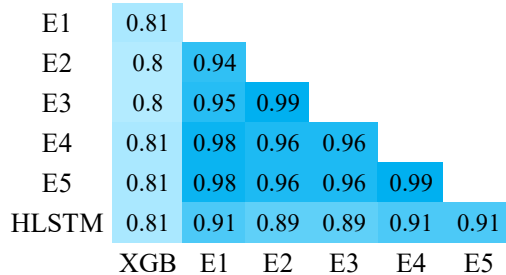


Figure 2: Pearson correlation matrix of a model to other models. E1–E5 denote ensemble models.

ELMo For the flat model, we pre-train ELMo by only fine-tuning the scalar-mix weights, as suggested in Peters et al. (2018). We extract a 1024-dimension bag-of-words representation for each turn and concatenate the three turns into a 3072-dimension vector which is passed to a multilayer perceptron (MLP). For the hierarchical model, we employ two methods: 1) run an LSTM model over each turn’s representation 2) pre-extract all three layer weights (LSTM and CNN) and concatenate them into a 3072-dimension vector representation for each turn, which is then passed to an LSTM model. We report the results of the latter pre-extracted method as it performs better.

BERT For the implementation details of $BERT_{BASE}$, we refer interested readers to Devlin et al. (2018). Note that for hierarchical BERT, we use a six-layer UTRS as the context encoder. Each layer of UTRS consists of a multi-head attention block with four heads, where the dimension of each head is set to be ten, and a convolution feed forward block with 50 filters. We use modified Adam optimizer from Devlin et al. (2018) to train our model. The initial learning rate and dropout are $5e-5$ and 0.3 respectively.

LSTM and Universal Transformer We train hierarchical LSTMs with hidden sizes of $\{1000, 1500\}$ using different dropouts $\{0.2, 0.3, 0.4, 0.5\}$. The best LSTMs (without additional features, with GLoVe, with GLoVe+Emo2Vec) reported in Figure 2 have a hidden size of 1000 and dropout of 0.5, a hidden size of 1500 and dropout of 0.2, and a hidden size of 1000 and dropout of 0.4 respectively. Then, we train the UTRS using the best hyper-parameters found by the GP. It has a hidden size of 488 with a single hop and ten attention multi-heads. Noam (Vaswani et al., 2017) is used as the learning rate decay.

Gaussian Processes GP hyper-parameter search returns a set of hyper-parameters, both continuous and discrete, and it returns the validation set F1 score. We implement the GP model using an existing library called GPyOpt.³ We run a GP for 100 iterations using the Expected Improvement (Jones et al., 1998) acquisition function with 0.05 jitter as a starting point. We use a hierarchical universal transformer (HUTRS) as the base model since is the model with the most hyper-parameters to tune with a single split.

3.2 Evaluation Metrics

The task is evaluated with a micro F1 score for the three emotion classes, i.e., Happy, Sad and Angry, and by taking the harmonic mean of the precision and the recall. This scoring function has been provided by the challenge organizers (Chatterjee et al., 2019b).

3.3 Voting Scheme

For each model, we randomly shuffle and split the training set ten times and we apply a voting scheme to create a more robust prediction. We use a majority vote scheme to select the most often seen predictions, and in case of ties, we give the priority to *Others*. This scheme is applied to all end-to-end models since it improved the validation set performance.

3.4 Ensemble Models

To further refine our predictions, we build ensembles of different models. We create five ensemble models by combining the hierarchical version of BERT, LSTM, and UTRS. Finally, we gather two lesser-performing models, a hierarchical LSTM and the best feature-based model (XGBoost with ELMo and DeepMoji features), and we combine them with five ensemble predictions using majority voting to get our final prediction. Finally, we show the Pearson correlation between models in Figure 2.

3.5 Experimental Results

From Table 1, we can see that the DeepMoji features outperforms all the other features by a large margin. Indeed, DeepMoji has been trained using a large emotion corpus, which is compatible with the current task. Emoji2Vec get a very low F1-score since it includes only emojis, and indeed,

³<http://sheffielddml.github.io/GPyOpt/>

by adding GLoVe, a more general embedding, we achieve better performance. For the end-to-end approach, hierarchical biLSTM with GLoVe word embedding achieves the highest score with a 75.64% F1-score. Our ensemble achieves a higher score compared to individual models. The best ensemble model achieves a 76.77% F1-score. As shown in Table 3, the ensemble method is effective to maximize the performance from a bag of models.

4 Related work

Emotional knowledge can be represented in different ways. Word-level emotional representations, inspired from word embeddings, learn a vector for each word, and have shown effectiveness in different emotion related tasks, such as sentiment classification (Tang et al., 2016), emotion classification (Xu et al., 2018), and emotion intensity prediction (Park et al., 2018). Sentence-level emotional representations, such as DeepMoji (Felbo et al., 2017a), train a biLSTM model to encode the whole sentence to predict the corresponding emoji of the sentence. The learned model achieves state-of-the-art results on eight datasets. Sentiment lexicons from Taboada et al. (2011) show that word lexicons annotated with sentiment/emotion labels are effective in sentiment classification. This method is further developed using both supervised and unsupervised approaches in Wang and Xia (2017). Also, other models, such as a deep averaging network (Iyyer et al., 2015), attention-based network (Winata et al., 2018), and memory network (Dou, 2017), have been investigated to improve the classification performance. Practically, the application of emotion classification has been investigated on interactive dialogue systems (Bertero et al., 2016; Winata et al., 2017; Siddique et al., 2017; Fung et al., 2018).

5 Conclusion

In this paper, we compare different pre-trained word embedding features by using Logistic Regression and XGBoost along with flat and hierarchical architectures trained in end-to-end models. We further explore a GP for faster hyper-parameter search. Our experiments show that hierarchical architectures give significant improvements and we further gain accuracy by combining the pre-trained features with end-to-end models.

References

- Dario Bertero, Farhad Bin Siddique, Chien-Sheng Wu, Yan Wan, Ricky Ho Yin Chan, and Pascale Fung. 2016. Real-time speech emotion and sentiment recognition for interactive dialogue systems. In *Proceedings of the 2016 Conference on Empirical Methods in Natural Language Processing*. pages 1042–1047.
- Ankush Chatterjee, Umang Gupta, Manoj Kumar Chinnakotla, Radhakrishnan Srikanth, Michel Galley, and Puneet Agrawal. 2019a. Understanding emotions in text using deep learning and big data. *Computers in Human Behavior* 93:309–317.
- Ankush Chatterjee, Kedhar Nath Narahari, Meghana Joshi, and Puneet Agrawal. 2019b. Semeval-2019 task 3: Emocontext: Contextual emotion detection in text. In *Proceedings of The 13th International Workshop on Semantic Evaluation (SemEval-2019)*. Minneapolis, Minnesota.
- Tianqi Chen and Carlos Guestrin. 2016. **XGBoost: A scalable tree boosting system**. In *Proceedings of the 22nd ACM SIGKDD International Conference on Knowledge Discovery and Data Mining*. ACM, New York, NY, USA, KDD '16, pages 785–794. <https://doi.org/10.1145/2939672.2939785>.
- Mostafa Dehghani, Stephan Gouws, Oriol Vinyals, Jakob Uszkoreit, and Łukasz Kaiser. 2018. Universal transformers. *arXiv preprint arXiv:1807.03819*.
- Jacob Devlin, Ming-Wei Chang, Kenton Lee, and Kristina Toutanova. 2018. Bert: Pre-training of deep bidirectional transformers for language understanding. *arXiv preprint arXiv:1810.04805*.
- Zi-Yi Dou. 2017. Capturing user and product information for document level sentiment analysis with deep memory network. In *Proceedings of the 2017 Conference on Empirical Methods in Natural Language Processing*. pages 521–526.
- Ben Eisner, Tim Rocktäschel, Isabelle Augenstein, Matko Bosnjak, and Sebastian Riedel. 2016. emoji2vec: Learning emoji representations from their description. In *Proceedings of The Fourth International Workshop on Natural Language Processing for Social Media*. pages 48–54.
- Bjarke Felbo, Alan Mislove, Anders Søgaard, Iyad Rahwan, and Sune Lehmann. 2017a. Using millions of emoji occurrences to learn any-domain representations for detecting sentiment, emotion and sarcasm. In *Proceedings of the 2017 Conference on Empirical Methods in Natural Language Processing*. pages 1615–1625.
- Bjarke Felbo, Alan Mislove, Anders Søgaard, Iyad Rahwan, and Sune Lehmann. 2017b. Using millions of emoji occurrences to learn any-domain representations for detecting sentiment, emotion and sarcasm. In *2017 Conference on Empirical Methods in Natural Language Processing Empirical Methods in Natural Language Processing*. Association for Computational Linguistics.
- Pascale Fung, Dario Bertero, Peng Xu, Ji Ho Park, Chien-Sheng Wu, and Andrea Madotto. 2018. Empathetic dialog systems.
- Sepp Hochreiter and Jürgen Schmidhuber. 1997. Long short-term memory. *Neural computation* 9(8):1735–1780.
- Chao-Chun Hsu and Lun-Wei Ku. 2018. Socialnlp 2018 emotionx challenge overview: Recognizing emotions in dialogues. In *Proceedings of the Sixth International Workshop on Natural Language Processing for Social Media*. pages 27–31.
- Mohit Iyyer, Varun Manjunatha, Jordan Boyd-Graber, and Hal Daumé III. 2015. Deep unordered composition rivals syntactic methods for text classification. In *Proceedings of the 53rd Annual Meeting of the Association for Computational Linguistics and the 7th International Joint Conference on Natural Language Processing (Volume 1: Long Papers)*. volume 1, pages 1681–1691.
- Donald R Jones, Matthias Schonlau, and William J Welch. 1998. Efficient global optimization of expensive black-box functions. *Journal of Global optimization* 13(4):455–492.
- Yanghoon Kim, Hwanhee Lee, and Kyomin Jung. 2018. Attnconvnet at semeval-2018 task 1: Attention-based convolutional neural networks for multi-label emotion classification. In *Proceedings of The 12th International Workshop on Semantic Evaluation*. pages 141–145.
- Bryan McCann, James Bradbury, Caiming Xiong, and Richard Socher. 2017. Learned in translation: Contextualized word vectors. In *Advances in Neural Information Processing Systems*. pages 6294–6305.
- Ji Ho Park, Peng Xu, and Pascale Fung. 2018. Plusemo2vec at semeval-2018 task 1: Exploiting emotion knowledge from emoji and# hashtags. In *Proceedings of The 12th International Workshop on Semantic Evaluation*. pages 264–272.
- 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. 2011. Scikit-learn: Machine learning in Python. *Journal of Machine Learning Research* 12:2825–2830.
- Jeffrey Pennington, Richard Socher, and Christopher Manning. 2014. Glove: Global vectors for word representation. In *Proceedings of the 2014 conference on empirical methods in natural language processing (EMNLP)*. pages 1532–1543.

- 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)*. volume 1, pages 2227–2237.
- Farhad Bin Siddique, Onno Kampman, Yang Yang, Anik Dey, and Pascale Fung. 2017. Zara returns: Improved personality induction and adaptation by an empathetic virtual agent. *Proceedings of ACL 2017, system demonstrations* pages 121–126.
- 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*. Association for Computational Linguistics, pages 1631–1642. <http://www.aclweb.org/anthology/D13-1170>.
- Maite Taboada, Julian Brooke, Milan Tofiloski, Kimberly Voll, and Manfred Stede. 2011. Lexicon-based methods for sentiment analysis. *Computational linguistics* 37(2):267–307.
- Duyu Tang, Furu Wei, Bing Qin, Nan Yang, Ting Liu, and Ming Zhou. 2016. Sentiment embeddings with applications to sentiment analysis. *IEEE Transactions on Knowledge and Data Engineering* 28(2):496–509.
- 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.
- Leyi Wang and Rui Xia. 2017. Sentiment lexicon construction with representation learning based on hierarchical sentiment supervision. In *Proceedings of the 2017 Conference on Empirical Methods in Natural Language Processing*. pages 502–510.
- Genta Indra Winata, Onno Kampman, Yang Yang, Anik Dey, and Pascale Fung. 2017. Nora the empathetic psychologist. *Proc. Interspeech 2017* pages 3437–3438.
- Genta Indra Winata, Onno Pepijn Kampman, and Pascale Fung. 2018. Attention-based lstm for psychological stress detection from spoken language using distant supervision. In *2018 IEEE International Conference on Acoustics, Speech and Signal Processing (ICASSP)*. IEEE, pages 6204–6208.
- Peng Xu, Andrea Madotto, Chien-Sheng Wu, Ji Ho Park, and Pascale Fung. 2018. Emo2vec: Learning generalized emotion representation by multi-task training. In *Proceedings of the 9th Workshop on Computational Approaches to Subjectivity, Sentiment and Social Media Analysis*. pages 292–298.