EmpTransfo: A Multi-head Transformer Architecture for Creating Empathetic Dialog Systems

Rohola Zandie and Mohammad H. Mahoor
Department of Electrical and Computer Engineering, University of Denver, USA
rohola.zandie@du.edu and mmahoor@du.edu

Abstract
Understanding emotions and responding accordingly is one of the biggest challenges of dialog systems. This paper presents EmpTransfo, a multi-head Transformer architecture for creating an empathetic dialog system. EmpTransfo utilizes state-of-the-art pre-trained models (e.g., OpenAI-GPT) for language generation, though models with different sizes can be used. We show that utilizing the history of emotions and other metadata can improve the quality of generated conversations by the dialog system. Our experimental results using a challenging language corpus show that the proposed approach outperforms other models in terms of Hit@1 and PPL (Perplexity).

Introduction
Humans have the unique capability to communicate with nuanced emotions through natural languages. Most of the existing conversational dialog systems focus on language understanding and improving the generated responses. Although these features are essential in building dialog systems, they lack empathetic features for conversation, which is essential for quality communication. To increase user satisfaction, dialog systems need to understand and incorporate emotions to respond with proper emotions. The positive effects of using emotional dialog systems also have been proved in many areas like customer satisfaction and healthcare applications (Dino et al. 2019).

Recent advances in Natural Language Processing (NLP) with the idea of using pre-trained models have led to remarkable results in different NLP tasks. Even though applying the same idea in dialog systems has resulted in improved models in terms of language understanding and generation, integrating other information like emotions and context knowledge is still challenging. To build empathetic conversational agents, machines need to have the ability to recognize and predict emotions based on the history of conversations and use them in interacting with users.

Corpora used in building most of the traditional dialog system include general conversations, although these datasets are usually large scale, they lack specificity and do not contain metadata such as emotions.

Copyright © 2020, Association for the Advancement of Artificial Intelligence (www.aaai.org). All rights reserved.
3. We design the model in a way that can be used with larger or smaller pre-trained models without changing the architecture of the system. This gives us the flexibility to use EmpTransfo in different settings based on our needs.

Related Work

Most of the work on utilizing emotions in conversational systems use large Twitter datasets (Sordoni et al. 2015) that contain emojis as the meta-information for the emotions. In (Zhou and Wang 2017), they use the Twitter dataset and apply a preprocessing method to create a conversational dataset with 64 different emojis that represent different emotions. In the preprocessing step, they used tweets and responses that contain at least one emoji and filtered other emojis based on the occurring frequency. They used a CVAE (Sohn, Lee, and Yan 2015) network to train and generate emotional responses. The choice of using emojis to represent emotion is noisy because it is too fine-grained and in many cases, the combination of different emojis hardly corresponds to any specific emotions.

In (Colombo et al. 2019), they used a seq2seq framework with vector representation for emotions, desired emotion, a regularizer to penalize neutral words and a sampling method that forces the generation of emotionally relevant words.

There are two papers on NLPCC dataset (Mi et al. 2014), Chinese language dataset with eight emotion categories. The first one is Emotional Chatting Machine (Zhou et al. 2018) that uses a seq2seq architecture with embedding of emotions along with words and an internal and external memory mechanism to generate emotional responses. The second work is EmoDS (Song et al. 2019) that uses a seq2seq approach with a training objective based on an emotion classifier that promotes implicit emotion generation. Both works use the lexical attention mechanism in decoder with more focus on emotional words to inject explicit emotions into responses.

Based on the reports from above-mentioned works, all the seq2seq based models tend to generate generic responses that can’t capture all the emotions equally well. Furthermore, all previous works use a machine annotating approach in the training process that introduces noise in results.

The most relevant work to ours is (Rashkin et al. 2019). They introduced a new dataset called EMPATHETICDIALOUGUES that contains meta-information about conversations. The meta-information is a label that shows emotion and also the situation in which the conversation has happened. They proposed two architectures, one retrieval-based model which looks for the best match using the BERT encoder (Devlin et al. 2018) and a generative model using Transformer architecture. Using one label for the whole conversation and not each utterance makes it harder for the models to find proper correlations.

On the other hand, recent progress has shown promising results on pre-trained language models on conversational models in chit-chat settings. Recently, (Wolf et al. 2019b) showed that using a fine-tuned GPT (Generative Pre-
\( U_i = \{t_1, t_2, ..., t_N\} \). Also, for each utterance, there is an emotion corresponding to it, resulting in the sequence of emotions \( E = \{e_1, e_2, ..., e_n\} \).

In our dataset, a sample is a sequence of utterances \( \{u_1, u_2, ..., u_{T-1}, u_{next}\} \) in which \( u_{next} \) can be either the correct next utterance \( u_T \) or a distractor from the set of distractors \( U'_T \). A distractor is a random utterance from the dataset. In the same way, if the corresponding sequence of emotions is \( \{e_1, e_2, ..., e_{T-1}, e_{next}\} \), then \( e_{next} \) is either the correct next emotion \( e_T \) or a distractor from the set of distractors \( E'_T \). A distractor emotion is a random emotion other than \( e_T \) from the set of all emotions.

Our model takes a sequence as input in the embedding space and passes it into a Transformer. The Transformer architecture \( \text{[Vaswani et al. 2017]} \) consists of a multi-layer Transformer decoder block. Each Transformer decoder block applies a masked multi-headed self-attention operation followed by a feedforward and a normalization layer over the input hidden states and gives the same size hidden states in the output. We then feed the output of the Transformer to three feed-forward linear heads, responsible for generating the next emotion, utterance, and token. Here, we used a 12-layer architecture but it can be extended or reduced to other model sizes. In the following, we define these three different heads and their corresponding loss functions.

1-Language modeling head: Language modeling is the task of predicting the next token given a sequence of tokens as the context. If we have a sequence of tokens for the correct next utterance as \( U_T = \{t_1, t_2, ..., t_N\} \), then the conditional probability of the next token is:

\[
P(t_i|t_1, ...t_{i-1}) = \text{softmax}(h * W_1)
\]

In which, \( h \) is the last hidden layer of the transformer model and \( W_1 \) is the token embedding matrix that is learned in training. Then we can define the loss function based on cross-entropy as:

\[
\mathcal{L}_1(U_T) = - \sum_{i=1}^N \log P(t_i|t_1, ..., t_{i-1})
\]

where the context of all previous tokens is encoded in a fixed dimension vector. It should be noted that the language modeling loss is not trained on the set of next utterance distractors \( U'_T \).

2-Next utterance prediction head: Following \( \text{[Devlin et al. 2018]} \), in the next utterance prediction, we try to train the model to predict the next utterance in the conversation. The model learns to distinguish between the correct next utterance among a set of random distractors from other parts of the dataset. More specifically, we create a classifier to calculate the probabilities of the next utterance:

\[
P_u(a|u_1, u_2, ..u_{T-1}) = \text{softmax}(h_1 * W_2)
\]

and \( a \) is defined as:

\[
a = \begin{cases} 1 & u_{next} = u_T \\ 0 & u_{next} \in U'_T \end{cases}
\]

\( h_1 \) is the hidden state for the last token from the Transformer decoder and \( W_2 \) is the weight matrix that is learned for the utterance prediction. Then, the loss function based on cross-entropy is:

\[
\mathcal{L}_2(U_{1:T}) = -\log P_u(a|u_1, u_2, ..u_{T-1})
\]

3-Next emotion prediction head: Similar to the next utterance prediction, the model is trained to distinguish between the correct next emotion among a set of distractors. The reason to add this head is to make the model learn not only the grammatical and language structure but also the appropriate emotions for any given history of utterances. We can define:

\[
P_e(e|e_1, e_2, ..., e_{T-1}) = \text{softmax}(h_{1-1} * W_3)
\]

where \( e \) represents:

\[
e = \begin{cases} 1 & e_{next} = e_T \\ 0 & e_{next} \in E'_T \end{cases}
\]

and \( h_{1-1} \) is the hidden state of one to the last token from the Transformer decoder and \( W_3 \) is the weights to be learned during the training for the emotion prediction task. The loss function for next emotion prediction is defined with cross-entropy:

\[
\mathcal{L}_3(U_{1:T-1}) = -\log P_e(e|e_1, e_2, ..e_{T-1})
\]

Finally, we optimize the following objective which is the total loss function:

\[
\mathcal{L}_{\text{total}} = c_1 \mathcal{L}_1 + c_2 \mathcal{L}_2 + c_3 \mathcal{L}_3
\]

where \( c_1, c_2, c_3 \) are hyperparameters that are tuned experimentally. In our experiments, we design the models with and without the “next emotion prediction” head for comparison.
Input Representation

We use DAILY DIALOG (Li et al. 2017) dataset, which is labeled with emotion and action tags per utterance, and with a topic tag for the whole conversation. In Table 2, a sample conversation in the preprocessed dataset is shown. A conversation is a sequence of utterances and each utterance can be more than one sentence, but the emotion and action information are defined per utterance. We also add distractors to each sample in the dataset. In the table, the row highlighted in red with the number d shows a distractor.

All the models use learned positional embeddings with a length of up to 512. Figure 3 demonstrates the input representation. The embeddings that have been used are:

1. **Token embedding**: The input sentences are tokenized using byte pair encoding (BPE) with a vocabulary size of 40,478.

2. **Emotion embedding**: Each one of seven emotions are considered as a special token to be learned as a new embedding. Emotion embeddings are copied for each token in the utterances and are added to the input of the network.

3. **Action embedding**: There are four actions for different communication functions that are used in dialog. The dialog acts are: Inform, Question, Directives, and Commissive. Dialog acts are also embedded with special tokens.

4. **Topics**: There are 10 topics defined in DAILY DIALOG that are specified for each conversation. For topics, we just concatenate topic embeddings to the beginning of the first input token embedding.

We sum all the embeddings and then feed them to the network.

**Training**

We used the OpenAI pre-trained model on BookCorpus dataset (Zhu et al. 2015) which covers more than 7,000 books. The books include narratives and dialogues, and emotions in a wide range of interactions between characters. This makes the pre-training suitable for the task of dialogue system training because it consists of sentences in a logical order without shuffling.

Starting with pre-trained weights, we fine-tune our model on the DAILY DIALOG dataset with the features mentioned in the Input Representation section. We use the combination of public evaluation and test as the validation set. After preprocessing the training set size is 76,502 and the validation size is 13,809.

We modify the dataset representation to cover different window positions of conversation history. Each sample in the modified dataset consists of the topic, last two utterances as history context, and the target utterance that can be either the real target or the distractor. The input window then moved forward to cover other parts of the conversation.

We fine-tuned the model with a batch size of 4 for a sequence length of 310 with 20 epochs over the training set of DAILY DIALOG dataset, this is about 1,500,000 steps. For the optimization of the loss function, we used Adam optimizer with a learning rate of 6.25e-5. Dropout rates for Transformer were borrowed from OpenAI GPT (Radford et al. 2018). All the proposed models are implemented in Pytorch using Transformers library (Wolf et al. 2019).

Here, we use the nucleus top-p sampling for language decoding (Holtzman et al. 2019). Given logits $u$ of the last hidden layer, and a sequence of $i-1$ tokens, $t_{1:i-1}$ as context, we have the following distribution over the next token $t_i$:

$$P(t_i = V_l | t_{1:i-1}) = \frac{\exp(u_l / T)}{\sum_l \exp(u_l / T)}$$

(10)

In which $V_l$ is the $l^{th}$ token in the vocabulary and $T$ is the temperature parameter. Higher values of $T$ result in more stochastic choices over tokens, though lower values of $T$ approach greedy and deterministic choices for the next token. Based on nucleus top-p sampling, we select $V^{(p)} \subset V$ as the smallest set such that

$$\sum_{V^{(p)} \subset V} P(t_i | t_{1:i-1}) \geq p$$

(11)

And then the distribution of Equation (10) should be rescaled to form a probability distribution. According to (Holtzman et al. 2019), $p = 0.9$ and $T = 0.7$ are closer to human text generation statistics and we use that for all our experiments.

**Results**

We evaluated our model on its ability to produce coherent and relevant utterances on the evaluation set. We also evaluate the proposed model on the task of generating emotional responses given the context on the evaluation set. Evaluation of the dialog systems can be done using automatic metrics and human evaluations. Here we restrict the evaluations to automatic metrics.

Table 1: A conversation in DailyDialog dataset

| # | Utterance                                      | Emotion    | Action    |
|---|-----------------------------------------------|------------|-----------|
| 1 | You look so happy, any good news?             | happiness  | question  |
| 2 | Yes, I’ve won the math contest                | happiness  | inform    |
| 3 | Really? Congratulations!                      | surprise   | question  |
| 4 | Thank you Paul.                               | happiness  | inform    |
| d | I really want to take him on my knee.         | anger      | inform    |

Table 2: Summary of the results on DAILY DIALOG evaluation set

| Model                              | Hit@1 | PPL ↓ | F1 ↑ | BLEU ↑ |
|------------------------------------|-------|-------|------|--------|
| Seq2Seq++Attention                 | 94.1  | 129.5 | 10.22| 5.87   |
| Transformer ranker                 | 17.20 | -     | 26.37| 15.79  |
| OpenAI GPT without emotion         | 75.01 | 10.19 | 18.2 | 3.755  |
| EmpTransfo                         | 77.25 | 10.63 | 19.39| 3.99   |
| EmpTransfo + topic                 | 76.87 | 10.23 | 18.37| 4.51   |
| EmpTransfo + action                | 77.73 | 9.17  | 18.86| 3.71   |
| EmpTransfo + action + topic        | 78.47 | 9.04  | 17.27| 2.45   |
| GPT (Radford et al. 2018)          | -     | -     | -    | -      |

https://github.com/roholazandie/EmpTransfo
Table 3: Examples of model responses.

| Input Prompt             | Model I finally passed all the exams! | Model I failed the exam | Model You scared me! |
|-------------------------|---------------------------------------|-------------------------|----------------------|
| Seq2Seq+Attention       | I’m sorry, but I’m not sure.          | I’m going to go to the same time. | I’m going to go to the job |
| Transformer Ranker      | How big was it?                       | You’re telling me! There are thousands of people here. | Come on! It is really a fun game. |
| OpenAI GPT w/o emotion | you look much better than before.     | let me take your place.  | what were you doing?  |
| EmpTransfo+action+topic | that’s great! you are really a genius. | Maybe you can try harder next time. | I’m so sorry. I thought you were not coming. |

Evaluation on coherence and relevance

**Baseline models:** For comparison, we used a seq2seq model with attention mechanism and a retrieval-based Transformer ranker dialog system as the baseline. The retrieval-based model is similar to [Rashkin et al. 2019](#), created in ParlAI framework [Miller et al. 2017](#).

Seq2Seq+Attention model uses linear attention with a hidden size of 128, learning rate of 0.01 and trained for 20 epochs. Transformer ranker uses a hidden size of 300 that is trained in 40 epochs with cross-entropy loss function. (all other hyperparameters are the defaults from the ParlAI repository).

**Evaluation metrics:** We use four different metrics to evaluate the models:

1. **Hit@1:** this metric is the accuracy of retrieving a gold next utterance among 19 random distractor responses sampled from other dialogues in the dataset.
2. **Perplexity (PPL):** perplexity is a measure of how well a language model predicts next tokens from the evaluation dataset. More specifically, it is the average per-token log probability over the evaluation set:
   \[
PPL(p) = e^{-\frac{1}{N} \sum_{w_i} \log(p_{w_i})}
   \] (12)
3. **BLEU:** it is a metric to measure the distance between machine-generated text with human golden labels [Papineni et al. 2002](#).
4. **F1 token:** measures the F1 score for token level comparison between the generated text and the golden labels.

In Table [2](#) we observe that all our proposed EmpTransfo models outperform baseline models in terms of Hit@1 and PPL. With more contextual features Hit@1 and PPL are improved. The proposed model also shows a significant improvement over [Shen et al. 2018](#) which has a PPL=23.8 over the same dataset. It also outperforms the Transformer retrieval-based model introduced in [Rashkin et al. 2019](#).

Transformer ranker model has a greater F1 and BLEU compared to other models which is expected because those metrics give higher scores for stronger similarity with the golden utterances in the datasets. BLEU is initially developed for machine translation and studies show that it is not a good metric for text generation evaluation [Sulem, Abend, and Rappoport 2018](#) and [Novikova et al. 2017](#). Table [2](#) also shows that adding more contextual information like topic and action results in higher Hit@1 and lower PPL. The reason behind this observation is that more contextual information provides the model with better information on selecting the correct next sentence and correct next token.

![Figure 4](https://example.com/figure4.png)

**Evaluation on emotion prediction**

In order to evaluate the next utterance emotion prediction, we calculate the precision and recall from the confusion matrix over the evaluation dataset. Figure 4 demonstrates the calculated confusion matrix with Precision=81.35, Recall=72.37 and F1=76.59. The proposed model achieves more than 3 percent improvement compared to [Chan and Lui 2018](#) whom report their best emotion prediction results with precision=70.81, recall=76.16 and F1=73.39. The confusion matrix also shows interesting observations such as the high rate of confusion between disgust and anger which has been observed in computer vision and facial expression recognition [Hasani and Mahoor 2017](#).

**Discussion**

This paper introduced EmpTransfo a multi-head Transformer model which is an empathetic aware dialog system to interact with users with higher quality in terms of coherence, relevance, and emotion. Our proposed method is built upon the state-of-the-art language model of OpenAI-GPT.
for language generation. One of the limitations in the proposed approach is requiring meta information of emotion, action, and topic in order to respond with the proper emotions.

**Conclusion and Future Work**

EmpTransfo is a scalable and fully data-driven neural conversational model that effectively exploits the information about emotion, action, and topic. It naturally combines conversational and non-conversational data through multi-task learning. This shows that multi-task learning for training conversational models is not only possible but necessary and can be extended to include more contextual data. As a future direction, we will use knowledge-base and other contextual information to develop knowledge-aware dialog systems.

**References**

[Chan and Lui 2018] Chan, Y. H., and Lui, A. K. F. 2018. Encoding emotional information for sequence-to-sequence response generation. In *2018 International Conference on Artificial Intelligence and Big Data (ICAIBD)*, 113–116. IEEE.

[Colombo et al. 2019] Colombo, P.; Witon, W.; Modi, A.; Kennedy, J.; and Kapadia, M. 2019. Affect-driven dialog generation. *arXiv preprint arXiv:1904.02793*.

[Devlin et al. 2018] Devlin, J.; Chang, M.-W.; Lee, K.; and Toutanova, K. 2018. Bert: Pre-training of deep bidirectional transformers for language understanding. *arXiv preprint arXiv:1810.04805*.

[Dino et al. 2019] Dino, F.; Zandie, R.; Abdollahi, H.; Schoeder, S.; and Mahoor, M. H. 2019. Delivering cognitive behavioral therapy using a conversational social robot. In *2019 IEEE/RSJ International Conference on Intelligent Robots and Systems (IROS)*, 2089–2095.

[Hasani and Mahoor 2017] Hasani, B., and Mahoor, M. H. 2017. Facial expression recognition using enhanced deep 3d convolutional neural networks. In *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition Workshops*, 30–40.

[Holtzman et al. 2019] Holtzman, A.; Buys, J.; Forbes, M.; and Choi, Y. 2019. Dailydialog: A manually labelled multi-turn dialogue dataset. *arXiv preprint arXiv:1904.09751*.

[Li et al. 2017] Li, Y.; Su, H.; Shen, X.; Li, W.; Cao, Z.; and Niu, S. 2017. Dailydialog: A manually labelled multi-turn dialogue dataset. *arXiv preprint arXiv:1710.03957*.

[Mi et al. 2014] Mi, C.; Yang, Y.; Wang, L.; Li, X.; and Dai, L. 2014. Detection of loan words in uygur texts. In Zong, C.; Nie, J.-Y.; Zhao, D.; and Feng, Y., eds., *Natural Language Processing and Chinese Computing*, 103–112. Berlin, Heidelberg: Springer Berlin Heidelberg.

[Miller et al. 2017] Miller, A. H.; Fung, W.; Fisch, A.; Lu, J.; Batra, D.; Bordes, A.; Parikh, D.; and Weston, J. 2017. Parlai: A dialog research software platform. *arXiv preprint arXiv:1705.06476*.

[Novikova et al. 2017] Novikova, J.; Dušek, O.; Curry, A. C.; and Rieser, V. 2017. Why we need new evaluation metrics for nlg. *arXiv preprint arXiv:1707.06875*.

[Papineni et al. 2002] Papineni, K.; Roukos, S.; Ward, T.; and Zhu, W.-J. 2002. Bleu: a method for automatic evaluation of machine translation. In *Proceedings of the 40th annual meeting on association for computational linguistics*, 311–318. Association for Computational Linguistics.

[Radford et al. 2018] Radford, A.; Narasimhan, K.; Salimans, T.; and Sutskever, I. 2018. Improving language understanding by generative pre-training.

[Rashkin et al. 2019] Rashkin, H.; Smith, E. M.; Li, M.; and Bourauel, Y.-L. 2019. Towards empathetic open-domain conversation models: A new benchmark and dataset. In *Proceedings of the 57th Conference of the Association for Computational Linguistics*, 5370–5381.

[Shen et al. 2018] Shen, X.; Su, H.; Niu, S.; and Dembreg, V. 2018. Improving variational encoder-decoders in dialogue generation. In *Thirty-Second AAAI Conference on Artificial Intelligence*.

[Sohn, Lee, and Yan 2015] Sohn, K.; Lee, H.; and Yan, X. 2015. Learning structured output representation using deep conditional generative models. In *Advances in neural information processing systems*, 3483–3491.

[Song et al. 2019] Song, Z.; Zheng, X.; Liu, L.; Xu, M.; and Huang, X.-J. 2019. Generating responses with a specific emotion in dialog. In *Proceedings of the 57th Annual Meeting of the Association for Computational Linguistics*, 3685–3695.

[Sordoni et al. 2015] Sordoni, A.; Galley, M.; Auli, M.; Brockett, C.; Ji, Y.; Mitchell, M.; Nie, J.-Y.; Gao, J.; and Dolan, B. 2015. A neural network approach to context-sensitive generation of conversational responses. *arXiv preprint arXiv:1506.06714*.

[Sulem, Abend, and Rappoport 2018] Sulem, E.; Abend, O.; and Rappoport, A. 2018. Bleu is not suitable for the evaluation of text simplification. *arXiv preprint arXiv:1810.05995*.

[Vaswani et al. 2017] Vaswani, A.; Shazeer, N.; Parmar, N.; Uszkoreit, J.; Jones, L.; Gomez, A. N.; Kaiser, Ł.; and Polosukhin, I. 2017. Attention is all you need. In *Advances in neural information processing systems*, 5998–6008.

[Wolf et al. 2019a] Wolf, T.; Debut, L.; Sanh, V.; Chaumond, J.; Delangue, C.; Moi, A.; Cistac, P.; Rault, T.; Louf, R.; Funtowicz, M.; and Brew, J. 2019a. Huggingface’s transformers: State-of-the-art natural language processing. *ArXiv abs/1910.03771*.

[Wolf et al. 2019b] Wolf, T.; Sanh, V.; Chaumond, J.; and Delangue, C. 2019b. Transfertransfo: A transfer learning approach for neural network based conversational agents. *arXiv preprint arXiv:1901.08149*.

[Zhang et al. 2018] Zhang, S.; Dinan, E.; Urbanek, J.; Szlam, A.; Kiela, D.; and Weston, J. 2018. Personalizing dialogue agents: I have a dog, do you have pets too? *arXiv preprint arXiv:1801.07243*.

[Zhou and Wang 2017] Zhou, X., and Wang, W. Y. 2017. Mojitalk: Generating emotional responses at scale. *arXiv preprint arXiv:1711.04090*.

[Zhou et al. 2018] Zhou, H.; Huang, M.; Zhang, T.; Zhu, X.; and Liu, B. 2018. Emotional chatting machine: Emotional
conversation generation with internal and external memory. In *Thirty-Second AAAI Conference on Artificial Intelligence*.

[Zhu et al. 2015] Zhu, Y.; Kiros, R.; Zemel, R.; Salakhutdinov, R.; Urtasun, R.; Torralba, A.; and Fidler, S. 2015. Aligning books and movies: Towards story-like visual explanations by watching movies and reading books. In *Proceedings of the IEEE international conference on computer vision*, 19–27.