Hybrid Models for Dialogue Generation and Q/A
Gator Generators

Venkata Sai Dheeraj Naganaboina, Brandon Silva, Tarun Chandra Narahari, Saikumar Padamati
{vnaganaboina,brandon.silva,tnarahari,padamatisaikumar}@ufl.edu

Abstract

Recent advances in natural language processing have led to the development of state-of-the-art models for question-answering (Q&A) and dialogue tasks. However, these models often struggle with handling complex, real-world scenarios, where multiple modalities, such as text, speech, and visual cues, are involved. Hybrid models, which combine the strengths of multiple models or modalities, have shown promise in addressing these challenges. This study aims to evaluate the performance of hybrid models in comparison to baseline state-of-the-art models for general Q&A and dialogue tasks using both automated and human evaluations.

1 Introduction

The primary objective of this research is to evaluate the performance of hybrid models for general Q&A and dialogue tasks in comparison to state-of-the-art models. Automated and human evaluation techniques will be used to assess the effectiveness of the models. Automated evaluations will use objective metrics such as ROUGE, BLEU score, and perplexity, while human evaluations will involve participants rating the models based on their overall quality, naturalness, and effectiveness.

This research aims to contribute to the existing literature by providing empirical evidence on the effectiveness of hybrid models for general Q&A and dialogue tasks. The findings of this study have implications for the development of more sophisticated NLP models that can handle real-world complex scenarios and improve overall performance.

We built a hybrid architecture that combines the strengths of transformers with its attention mechanism and LSTM's long-term dependency. This paper provides information about the performance of fine-tuned baseline models and also the architecture of the hybrid model which is the combination of LSTM and transformer. In the future works section, we also provide some more ideas to improve the performance of this hybrid model.

2 Related Work

Hybrid models are being researched in many domains, including natural language processing, computer vision, forecasting, and more. With the rise in popularity of Transformer models, many hybrid approaches that utilize Transformers have been developed as well. A recent example of this is the Block-Recurrent Transformer (Hutchins et al., 2022). This model incorporates LSTM-style gates that apply a Transformer layer recurrently along a sequence, which operates on "blocks" of tokens, rather than token by token. These new Block-Transformer layers incorporate the gates for both cross and self-attention, leveraging the ability of LSTM cells to learn long-term information and attention that learns relationships among tokens.

Convolutional Neural Networks have also been used to extract features from token embeddings, which can be passed into a Transformer or RNN model. One example of using this technique involves using a CNN model for feature extraction at a character level, combined with a more conventional feature extraction using Word2Vec or FastText (Salur and Aydin, 2020). This process combines features from both an RNN model and CNN model for sentiment analysis, showing improved performances compared to single model approaches.

3 Baseline Architectures

In order to establish a proper baseline to compare our hybrid approaches, we tested a variety of state-of-the-art models for NLP tasks, including Transformers, LSTMs, and GANs. While the GAN architecture did not perform well on the tasks, the Transformers and LSTM networks did, which we used to develop our hybrid approaches.

3.1 Long-Short Term Memory Networks

Long Short-Term Memory Networks commonly called LSTM are a special type of Recurrent Neural Network that is used to address the Vanishing Gradient problem and to solve the long-term dependency problem. LSTM network consists of three gates Input gate, output gate, and Forget gate to control the flow of data in a memory cell. The Input gate controls the amount of new data that is entering a cell, the output gate controls the amount of information that is sent from a cell whereas the forget gate determines the amount of information that is retained or forgotten from the previous time step. LSTM has achieved significant success in various domains, such as speech recognition, natural language processing, and image captioning. These networks are especially advantageous for tasks that involve sequential data processing, and they excel at managing long-term dependencies in the data.

3.2 Transformers

Since the original release of the Transformer architecture in 2017 (Vaswani et al., 2017), a variety of novel Transformer models have been developed. Especially in the NLP domain, Transformers have shown to be highly effective at processing language. In order to provide an accurate comparison of performance between singleton and hybrid models, we compare multiple Transformer model baselines to our hybrid model approaches.

3.2.1 Open Pre-trained Transformer (OPT)

The OPT model was created as a response to many large language models being closed source, with no access to pretrained weights or code (Zhang et al., 2022). This model is a large Transformer model, trained with a maximum of 175B parameters, comparable to GPT3. For our purposes, we fine tune a pretrained model with 350M parameters on our datasets due to constraints of hardware. The model boasts a smaller training time compared to other large language Transformer models and high performance. The model is capable of few shot learning for a variety of NLP tasks such as sentiment analysis, toxity evaluations, and more.

3.2.2 BERT

The Bidirectional Encoder Representations from Transformers (BERT) model is a language model that is designed to have deep bidirectional representations of text during pretraining to allow the model to perform a variety of tasks by simply changing the head of the model (Devlin et al., 2019). The model uses masked input during training, where input tokens are randomly masked and the model must fill in the gaps while also generating its output. It can be used as both an encoder for next word prediction and a Seq2Seq model for generative output (such as summarization). The BERT model is a widely used baseline for language models trained on NLP tasks, which will help use evaluate the performance of our hybrid models.

3.2.3 GPT2

OpenAI's iterations of the GPT architecture have shown impressive results. GPT2 is quite outdated compared to it's newest iteration, GPT4, it still performs extremely well on a variety of NLP tasks, including dialogue and q&a (Radford et al., 2018). GPT2 is extremely common in a vartiety of applications, and widely known, making it an ideal baseline comparsion for our hybrid approaches.

4 Hybrid Architecture

The main aim of having a hybrid architecture is that it can address the disadvantages of existing stand-alone models. The following architecture is a combination of LSTM and Transformer.

4.1 LSTM - Transformer

With multi-head attention, transformers capture the dependencies between tokens within their own head. The idea behind adding an LSTM layer to this architecture is to accumulate the information gained by all the tokens with attention and pass it to the future tokens that help them in predicting the next token more accurately. Adding an LSTM layer also addresses the vanishing gradients problem in a deep transformer.

In this architecture, we made use of multiple decoder-based transformer blocks with self-attention and an LSTM block to accumulate their information. The LSTM block is placed in between the transformer blocks so that the calculated information can be accumulated once and then the calculations can continue. This helped in bringing down the loss considerably. The whole architecture is implemented using PyTorch. The current architecture contains 4 transformer blocks followed by an LSTM block followed by another 4 transformer blocks and an LSTM block and ending with 2 transformer blocks. So, the model has a total of 12 blocks, 10 transformer blocks, and 2 LSTM blocks and has more than 15M parameters in total. As the

architecture is deep, we used residual connections and dropouts in the transformer blocks to avoid the problem of vanishing gradients. Each transformer block has a multi-head attention mechanism followed by a feed-forward block made up of dense layers.

5 Datasets

In order to obtain a more comprehensive understanding of how hybrid models compare to baseline architectures, we conducted evaluations on three separate datasets. The first dataset is the Cornell Movie Dialogue Corpus, which involves generating responses to movie-related dialogue. The second dataset is the Ask Reddit QA dataset, which involves answering questions based on discussions from the Reddit website. The third dataset is the Stanford QA (SQuAD) dataset, which involves answering questions based on passages of text. By evaluating the models on these diverse datasets, we can gain a better understanding of their overall performance across a range of different tasks and scenarios.

5.1 Cornell Movie Dialogue Corpus

This corpus contains a collection of over 200,000 conversational exchanges between movie characters, taken from over 600 movies. The dialogues span a variety of genres, including drama, comedy, action, and horror.

The dataset is organized into a series of text files, with each file containing the text for a single movie.The text file used in this study contained a corpus of conversational exchanges between characters in movies. Each line of the file represented a single exchange and was formatted as follows: <character name> <character name> <text>. The dataset included a total of 220,579 conversational exchanges between 10,292 pairs of movie characters. These exchanges involved 9,035 unique characters from a total of 617 movies. Overall, the dataset included 304,713 utterances, providing a rich source of dialogue data for use in our analysis.

To prepare the raw text data for modeling, we carried out a series of preprocessing steps. The first step was to load the text file into a pandas dataframe, which allowed us to organize and manipulate the data in a convenient and efficient manner. We then performed text cleaning, which involved removing any unwanted characters, symbols, or formatting that could potentially interfere with the accuracy of the models. This cleaning step was crucial for ensuring that the text data was in a suitable format for further analysis.

Once the text was cleaned, we performed tokenization, which involved breaking the text into individual words or tokens. Tokenization is a common technique used in natural language processing to facilitate the analysis and manipulation of text data. We used a tokenization algorithm to split the text into individual tokens, which we then fed into the subsequent stages of the modeling process.

Finally, we used GloVe embedding to convert the text tokens into vector representations that capture their meaning and context. GloVe embedding is a popular technique used in natural language processing that maps words or tokens to vector representations based on the co-occurrence of words in a corpus of text. The resulting vector representations capture the semantic relationships between words and enable the models to understand the meaning and context of the text data. By performing these preprocessing steps, we were able to transform the raw text data into a format that was suitable for use in the subsequent modeling stages.

5.2 Ask Reddit QA

A question and answer corpus from the /r/askreddit subreddit. Designed for training seq2seq neural networks. The total corpus is 4,976,760 question and answer pairs.

In the data processing pipeline, the first step involved data cleansing, which aimed to remove any inaccuracies or inconsistencies in the dataset. Following this, a sorting process was applied to rank the questions based on user ratings. Next, multiple answers to the same question were combined and aggregated, and a new file was created with these combined answers. Finally, tokenization was performed as a preprocessing step to transform the text data into numerical vectors that can be used for machine learning algorithms. These steps were executed in a systematic manner to ensure the quality of the data and optimize the subsequent analysis.

5.3 Stanford QA

Stanford Question Answering Dataset (SQuAD) is a dataset created by researchers at Stanford University that is designed for machine learning tasks related to reading comprehension and question answering. It contains over 100,000 question-answer pairs that are based on Wikipedia articles, and each question in the dataset is associated with a specific

paragraph from a Wikipedia article that contains the answer to the question.

The dataset is organized into two parts: SQuAD 1.1 and SQuAD 2.0. SQuAD 1.1 is the original version of the dataset, while SQuAD 2.0 includes additional questions that do not have explicit answers in the associated paragraphs.

For data processing we provide JSON file as input and convert it into a pandas DataFrame. The JSON file is assumed to have a specific structure, with a root tag containing a 'title' tag and a list of 'paragraphs'. Each paragraph contains a 'context' tag and a list of 'qas'. Each 'qa' contains a 'question' tag, an 'id' tag, and a list of 'answers'. Each 'answer' contains a 'text' tag. We first open the JSON file and loads its contents using the json library. Then, it creates empty lists to store the values for context, question, and text. It loops through each paragraph in the JSON file, extracting the title and context tags, and then loops through each 'qa' in the paragraph, extracting the question, id, and answers tags. Finally, it loops through each answer in the 'answers' list, extracting the text. For each of these, it appends the values to the corresponding lists. After creating the lists, the function creates a new pandas DataFrame with columns named 'Context', 'Question', and 'Answer'. It populates the DataFrame with the values from the lists and drops any duplicate rows. Finally, it returns the resulting DataFrame.

6 Experiments

We implemented a series of models on Cornell Movie Corpus, Stanford Question Answering, AskReddit datasets to perform Dialogue Generation and Question answering tasks. We performed tests on LSTM, BERT, Longformer, GPT, OPT, and a hybrid model consisting of LSTM and Transformers. We started our experiments by preprocessing the data which includes data cleaning and formatting the data into a suitable format. Then we performed NLP techniques like Tokenization, Stop word removal, Normalization. We then padded the data to make them all equal length, after that we embedded the data using Glove embedding and Keras Embedding to make them into vectors. After the preprocessing, we trained our data with respective models, and we fine-tuned the hyperparameters to get the optimal result. We used perplexity and cross entropy loss as the automated evaluation metrics, while also performing human evaluation on the models.

Coming to our hybrid model, we have tried many architectures for getting the optimal loss. Firstly, we tried to add the LSTM layer at the end of four transformer blocks, but we haven't found much success in this. So, we tried to tweak it a little by changing the LSTM layer into middle so that we can accumulate the information after some computation based on attention is done. Then we sent the accumulated information through few more transformer blocks. We got better results by this approach than the first approach. We tried another approach by making the network deeper to decrease the loss. In our final architecture, we first added 4 transformer blocks then an LSTM layer. After that LSTM layer we added another 4 transformer blocks and another LSTM block to it. In the end we added another 2 transformer blocks and thus making this a deeper network so that it can learn easily and perform better. Totally, there are 10 transformer blocks and 2 LSTM layers in our hybrid model.

7 Results

Below are our model metrics for Cornell Movie Corpus,

Model	BLEU	Rouge	Perplexity
LSTM	0.21	0.19	20.7
BERT	0.19	0.17	20.1
OPT	0.385	0.3	19.2
GPT2	0.25	0.19	19.9
Hybrid Model	0.21	0.2	21.2

Below are our model metrics for Stanford Question Answering Dataset,

Model	BLEU	Rouge	Perplexity
LSTM	0.28	0.29	27
BERT	0.35	0.22	34.5
OPT	0.4	0.31	29.5
GPT2	0.3	0.23	16.1

Below are our model metrics for AskReddit Dataset,

Model	BLEU	Rouge	Perplexity
BERT	0.23	0.26	26.5
OPT	0.32	0.38	21.2
GPT2	0.29	0.25	24.1
Hybrid Model	0.24	0.22	26.4

8 Future Work and Discussion

With only around 15M parameters, our hybrid model was able to perform reasonably when compared to the pre-trained baseline models. Our hybrid model is implemented using only decoder blocks and a self-attention mechanism as the task is predicting the next word based on the input. Even in this setting, this model is able to generate coherent responses with proper syntax, but there is some problem with context. This model can be improved by adding cross-attention which would make it a better sequence-to-sequence generation model. We also see scope in training this model on a large corpus of text documents first and then fine-tuning the model based on the task at hand, since baseline Transformer models tested were fine-tuned from pre-trained weights.

One observation we made while implementing this model is that we can further increase the complexity of the hybrid models by adding reinforcement learning techniques to enhance model training, which can result in better performance overall. With increased complexity, the model would be able to perform better than our experiments and better contextualize input sequences.

9 Conclusion

We started this project as a comparative study between hybrid models and baseline models in dialogue generation and question answering and found out that using multiple features from different models would significanty improve the output of the model. The best example is ChatGPT which is a combination of GPT, and Proximal Policy Optimization (PPO). Accumulating the attention-based information helped a lot while generating the next token and this technique would work even better with long sequence inputs. As the field of natural language processing continues to evolve, it is likely that hybrid models will play an increasingly important role in text generation and other related tasks.

References

Jacob Devlin, Ming-Wei Chang, Kenton Lee, and Kristina Toutanova. 2019. Bert: Pre-training of deep bidirectional transformers for language understanding.

DeLesley Hutchins, Imanol Schlag, Yuhuai Wu, Ethan Dyer, and Behnam Neyshabur. 2022. Block-recurrent transformers.

Alec Radford, Jeffrey Wu, Rewon Child, David Luan, Dario Amodei, and Ilya Sutskever. 2018. Language models are unsupervised multitask learners.

Mehmet Umut Salur and Ilhan Aydin. 2020. A Novel Hybrid Deep Learning Model for Sentiment Classification. *IEEE Access*, 8:58080–58093.

Ashish Vaswani, Noam Shazeer, Niki Parmar, Jakob Uszkoreit, Llion Jones, Aidan N. Gomez, Lukasz Kaiser, and Illia Polosukhin. 2017. Attention is all you need.

Susan Zhang, Stephen Roller, Naman Goyal, Mikel Artetxe, Moya Chen, Shuohui Chen, Christopher Dewan, Mona Diab, Xian Li, Xi Victoria Lin, Todor Mihaylov, Myle Ott, Sam Shleifer, Kurt Shuster, Daniel Simig, Punit Singh Koura, Anjali Sridhar, Tianlu Wang, and Luke Zettlemoyer. 2022. Opt: Open pre-trained transformer language models.

10 Appendices

The table below is the results showing the outputs of our models with the ground truth for given questions.

Model	Question	Ground Truth	Model Answer
LSTM	What part of France were the Normans located?	the north	north of france
LSTM	Who was the Normans' main enemy in Italy, the Byzantine Empire and Armenia?	the Pechenegs, the Bulgars, and especially the Seljuk Turks	the pechenegs the bulgars
BERT	Hello, how are you doing today?	-	I've had a very good week, and I'm very much enjoying a lot of my time.
BERT	What part of France were the Normans located?	the north	in the north
OPT	What is the area of a triangle?	one half times the base times the height	base times height
OPT	What is the meaning of life?	-	Life is a term that describes the way the universe works. It usually comes from Plato, or from the Epicurean philosophers.
GPT 2	Hello, how are you doing today?	-	I'm still doing so well. My wife, the only one who's been here for the past two or three days, she's still here.
GPT 2	How old is the Queen of England?	she is ninety six years old	ninety six
Hybrid Model	Who won the FIFA world cup in 2022?	Argentina won the recent world cup	Argentina is the world cup win
Hybrid Model	What is the meaning of life?	-	Life depends on experiences and in how you do live