
Reinforcement Learning and Discourse Mutual Information based methods for Dialogue Generation and Evaluation

*Submitted in partial fulfilment
of the requirements of the
degree of*

Master of Technology

in

Visual Information and Embedded Systems

Submitted by

Arpit Dwivedi

Roll No: 17EC35005

Under the joint guidance of

Prof. Pawan Goyal and Prof. Arijit De



**Department of Electronics and Electrical
Communication Engineering**

Indian Institute of Technology Kharagpur

Kharagpur, West Bengal, India – 721 302

Academic Year 2021-22

**Department of Electronics and
Electrical Communication Engineering**
Indian Institute of Technology, Kharagpur

Certificate

This is to certify that this is a bonafide record of the project presented by

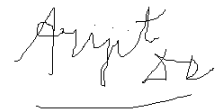
Arpit Dwivedi(17EC35005)

during Academic Year 2021-22 in partial fulfilment of the requirements of the
degree of Master of Technology in Visual Information and Embedded Systems.



Prof. Pawan Goyal

(Project Guide)



Prof. Arijit De

(Project Guide)

Date: 01/05/2022

ABSTRACT

Dialogue systems are becoming increasingly popular these days. Chatbots are being used by several organisations these days to handle queries across various fields. Therefore it becomes extremely important to provide appropriate responses to such queries that are not only grammatically correct but also make logical sense. Conventional generative dialogue systems trained using cross entropy loss suffer from several drawbacks such as degeneracy in the response quality due to incorrect penalisation of appropriate responses during training. To tackle this problem we propose a new training methodology in this thesis using reinforcement learning and retrieval dialogue systems. We analyse their improvement and compare their performance to the baseline models using several different metrics.

Evaluating dialogue systems is another topic of huge research interest. The standard word overlap based metrics do not yield the correct results since dialogues are open ended and there can be many possible responses for a given context. Moreover reliance on the ground truth is not desirable as it's availability may pose a problem especially in an online setting. So we propose a novel unreferenced metric for dialogue evaluation in this thesis that uses a model pre-trained specifically for dialogue understanding and compare its performance with the existing baselines by measuring the correlation with human judgements.

TABLE OF CONTENTS

ABSTRACT	3
TABLE OF CONTENTS	4
INTRODUCTION	5
CHAPTER-1: RL BASED TRAINING OF GENERATIVE DIALOGUE SYSTEM	6
LITERATURE REVIEW	6
THEORY	7
GENERATIVE MODEL ARCHITECTURE	7
RETRIEVAL MODEL ARCHITECTURE	9
ESIM	9
BERT	11
POLICY BASED REINFORCEMENT LEARNING	13
EXPERIMENTS	14
DATASET	14
TRAINING	14
VALIDATION	15
INFERENCE	16
EVALUATION AND RESULTS	18
RETRIEVAL MODELS	18
GENERATIVE MODELS	19
ABLATION STUDY	20
CHAPTER-2: DMI BASED EVALUATION OF DIALOGUE SYSTEMS	23
LITERATURE REVIEW	23
THEORY	24
DMI SCORE	24
DISCOURSE MUTUAL INFORMATION BASED TRAINING OF MODEL	24
ARCHITECTURE	25
PROPOSED EVALUATION METRIC	25
EXPERIMENTS	26
BASELINES	26
InferSent	26
DistilBERT-NLI	26
RUBER	27
MAUDE	27
CORRELATION WITH HUMAN JUDGEMENTS	27
EVALUATION AND RESULTS	28
CONCLUSION AND FUTURE WORK	29
REFERENCES	30

INTRODUCTION

The Transformer based models disrupted the field of natural language processing across multiple domains with Dialogue Generation being one of them as well. The authors of Attention Is All You Need [1] proposed the very first Seq2Seq model using a transformer. Since its inception, several transformer based models have shown to perform exceedingly well in the domain of dialogue generation. The convention followed for training such dialogue generation models were however based on supervised training using an input to the encoder and calculating the loss against an expected output or the golden response using Cross Entropy or similar loss function. This however might lead to degeneration of the response quality and moreover the responses are confined within a small space due to the nature of the algorithm used for training. This is because the cross entropy based methods might end up penalising appropriate responses just because they do not match with the target response. Moreover it also might be possible that the response might have been rephrased or some word might have been replaced by its synonym and the model ended up penalising that response. We propose a novel training algorithm for a transformer based Seq2Seq dialogue system using the Reinforce algorithm with a retrieval based model used for reward generation and allocation among tokens. Such training allows the model to learn a diverse set of responses as the retrieval model is capable of providing a goodness score for each response on a continuous scale. During inference we generate the response for the given context and evaluate the quality of responses using several metrics which clearly indicate an improvement in the quality of generated responses as compared to the existing cross-entropy based training methods.

Evaluation of dialogues is a very crucial task. Most of the responses may not exactly match with the ground truth and yet might be the perfect response for the given context. So we need to come up with ways to correctly evaluate the responses given a certain context. The standard word overlap metrics such as BLEU and METEOR are clearly not the right choice nor are other referenced metrics such as ROUGE and ADEM due to their reliance on the ground truth response. Unreferenced metrics like MAUDE, RUBER (partly) did make some progress by removing the reliance on ground truth label and scoring on the basis of the relationship between the context and response. However even the best of these metrics used standard pre-trained language models that are unable to capture the intricate details of dialogues. Therefore we propose to use a model that is pre-trained using the Discourse Mutual Information (DMI) based loss function and is known to perform exceedingly well in dialogue understanding tasks as an evaluation metric. To gauge the quality of the metric we compare its performance with other baselines by measuring its correlation with human judgements.

CHAPTER-1: RL BASED TRAINING OF GENERATIVE DIALOGUE SYSTEM

LITERATURE REVIEW

Conventional Seq2Seq architectures based on **RNNs** used the encoder to develop a fixed size representation of the given input sentence which was then passed to the decoder in order to generate the output sequence. However the primary problem with this architecture was that it suffered from an information “bottleneck”. It was unable to perform upto the mark for particularly long sentences since they were compressed to a fixed length vector.

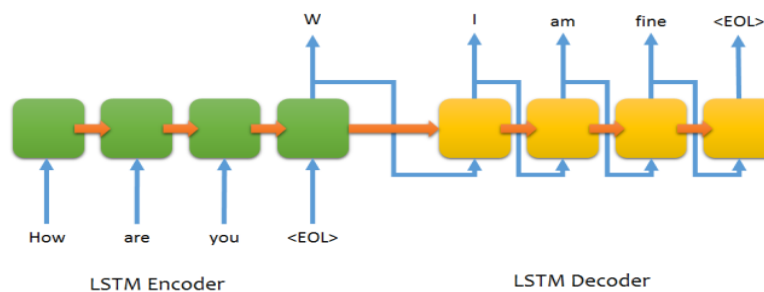


Figure 1:- LSTM based Seq2Seq model without attention [[Link](#)]

An important development came around with the development of Seq2Seq models with **attention** [2]. Attention is weighing individual words in the input sequence according to the impact they make on the target sequence generation. As a result now at every time step the decoder could attend to all the encoder hidden states to determine the output at that time step. This worked well but the primary issue remained the sequential computation performed by the RNNs. Due to this parallelization was not possible and this paved the way for transformers.

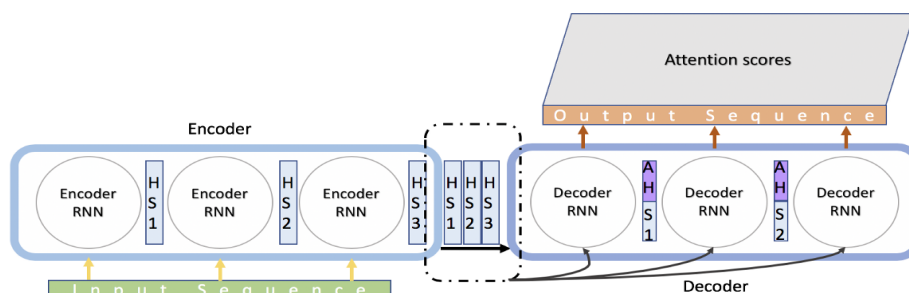


Figure 2:- RNN based Seq2Seq model with attention [[Link](#)]

THEORY

GENERATIVE MODEL ARCHITECTURE

Transformers proposed by Vaswani et. Al. in 2018 sparked a revolution in the field of Natural Language Processing. They got rid of the sequential computations by performing extensive parallelization. The architecture can be summarized as shown in the below figure.

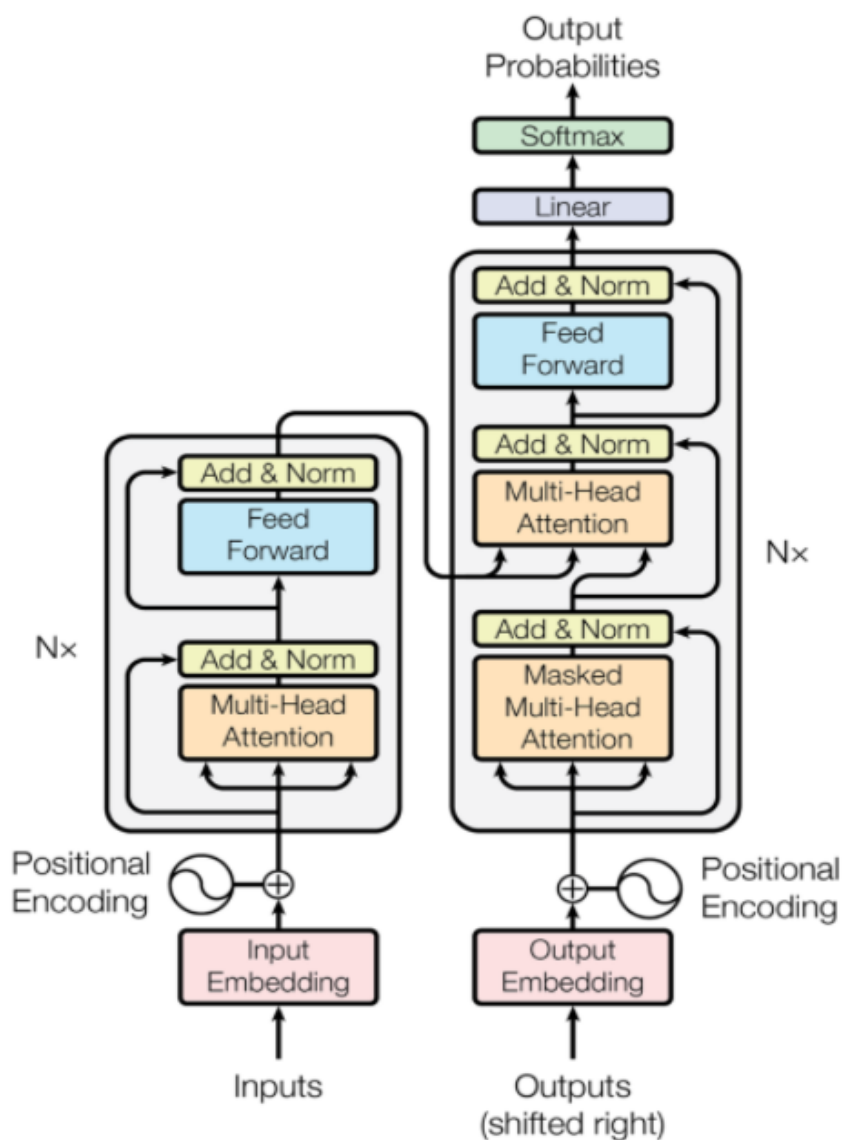


Figure 3:- Transformer architecture [\[1\]](#)

The encoder is shown on the left while the decoder is shown on the right. Both encoder and decoder consist of multiple modules stacked upon each other. First the input is encoded using a tokenizer. Then the positional encoding is added to indicate the relative position of each token in the input. This is particularly important as it is important to understand the relative placement of tokens since we are not using recurrent models any longer. Each module primarily consists of Multi-Headed Attention and Feedforward layers. Suppose Q, K, V and d_k refer to the query, key, value and the hidden dimension then the attention formula is given as follows:-

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

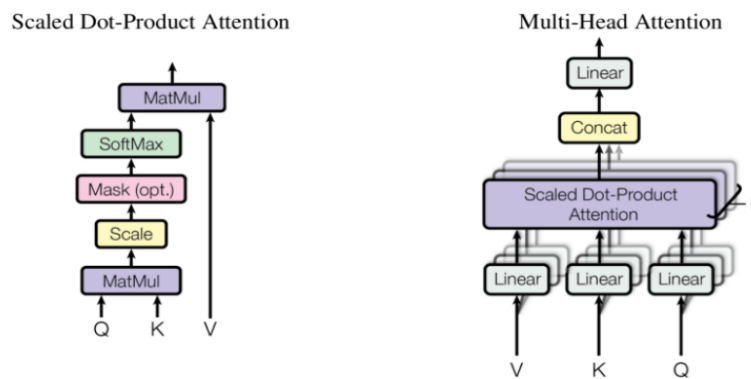


Figure 4:- Multi-headed attention in transformers [1]

There are 3 types of attention units that can be seen in the above figure. The first two are self attention within the encoder and decoder itself. This is done so that a token can have a look at the other tokens in the sequence. However masking is done in the case of decoder to prevent peeking ahead. The third type of attention is encoder-decoder attention which is done so that the target sequence can pay attention to the source sequence

The diagram on the right shows how this attention mechanism can be parallelized into numerous processes that can be used simultaneously. With linear projections of Q, K , and V , the attention mechanism is repeated several times. This allows the system to learn from a variety of Q, K , and V representations, which is useful to the model.

A pointwise feed-forward layer follows the multi-attention heads in both the encoder and decoder. This small feed-forward network has identical parameters at each position, which can be described as a separate, identical linear transformation of each element from the given sequence

RETRIEVAL MODEL ARCHITECTURE

We primarily use two retrieval models for training our generative models. They are ESIM and BERT. We will now discuss each of them in detail.

ESIM

The Enhanced Sequential Inference Model was originally proposed by Chen et. al. in 2016 [3] for Natural Language Inferencing. However, later work by Chen et.al. in 2019 [4] showed that ESIM performs equally well in the response selection task.

The Enhanced Sequential Inference Model has three main components. They are input encoding, local matching and matching composition.

- 1) **Input encoding**:- Similar to every Natural Language Processing model, the input to this model is given in the form of embedding vectors. Thus for a context $\mathbf{c} = [\mathbf{c}_1, \mathbf{c}_2, \dots, \mathbf{c}_n]$ and a response $\mathbf{r} = [\mathbf{r}_1, \mathbf{r}_2, \dots, \mathbf{r}_m]$ the input to the model will be $\mathbf{E}(\mathbf{c}) = [\mathbf{E}(\mathbf{c}_1), \mathbf{E}(\mathbf{c}_2), \dots, \mathbf{E}(\mathbf{c}_n)]$ and $\mathbf{E}(\mathbf{r}) = [\mathbf{E}(\mathbf{r}_1), \mathbf{E}(\mathbf{r}_2), \dots, \mathbf{E}(\mathbf{r}_m)]$ respectively where $\mathbf{E}(t)$ is an embedding vector representation for a token t using some embedding \mathbf{E} . In our case, we use the pretrained Blender-3B embeddings. These embedded tokens are then passed through a BiLSTM to get their contextual representation i.e., the earlier embeddings did not take the surrounding context of the token into account. To obtain that a Bi-LSTM is used similar to the idea proposed in ELMo [8]. Suppose i and j denote the i th and the j th token of the context and the response respectively after passing them through the first BiLSTM layer then we get:-

$$\begin{aligned} c_i^s &= \text{BiLSTM}_1(\mathbf{E}(\mathbf{c}), i), \\ r_j^s &= \text{BiLSTM}_1(\mathbf{E}(\mathbf{r}), j), \end{aligned}$$

- 2) **Local Matching**:- The next component includes local matching between the tokens of the context and the response. This is typically important when we want to find relation between context and response, i.e, in this case, whether the response is the correct next utterance for the given context. The relation can be measured through local semantic relation at the token level. This is done by cross attention between the context and the response tokens. The idea is to obtain a new representation of a context token using a weighted sum of the current representation of the response tokens and vice versa.

$$e_{ij} = (c_i^s)^T r_j^s .$$

The attention values \mathbf{e} are obtained using the attention matrix hence the name “Cross attention”. These attention matrix values are then passed through a Softmax layer to obtain attention weights which are then used to take a weighted sum of the response(context) tokens to find the new representation or the dual representation of the context (response) tokens.

$$\alpha_{ij} = \frac{\exp(e_{ij})}{\sum_{k=1}^n \exp(e_{ik})} , \quad c_i^d = \sum_{j=1}^n \alpha_{ij} r_j^s ,$$

$$\beta_{ij} = \frac{\exp(e_{ij})}{\sum_{k=1}^m \exp(e_{kj})} , \quad r_j^d = \sum_{i=1}^m \beta_{ij} c_i^s ,$$

To measure semantic similarity between response and context tokens, both the contextual and dual representations of a token are used to finally obtain the local matching vectors. In the figure shown below F is a one layer Feed-Forward layer with ReLU activation to reduce the dimension. The concatenation, difference and element wise multiplication between the two representations is used as a heuristic matching approach

$$c_i^l = F([c_i^s; c_i^d; c_i^s - c_i^d; c_i^s \odot c_i^d]) ,$$

$$r_j^l = F([r_j^s; r_j^d; r_j^s - r_j^d; r_j^s \odot r_j^d]) ,$$

- 3) **Matching composition**:- The next component is meant for composition of the local matching vectors. It composes the matching between the context and the response to finally determine the correctness of the response. Another layer of BiLSTM is used for composing local matching vectors yielding the composed vectors

$$c_i^v = \text{BiLSTM}_2(c^l, i) ,$$

$$r_j^v = \text{BiLSTM}_2(r^l, j) .$$

Pooling is used to extract a fixed size vector representation from the composed vectors which is then passed through an MLP Classifier with one hidden layer, tanh activation and a final softmax layer to get the output class label. Cross Entropy Loss was minimized to train the ESIM.

$$y = \text{MLP}([c_{max}^v; c_{mean}^v; r_{max}^v; r_{mean}^v]) .$$

The output of the ESIM is a binary-class label assigned w.r.t the given context-response pair with 1 indicating the response is fit to be the next utterance of the context while 0 implies it isn't. Once the ESIM had been trained, we used the class-1 probability from the learned ESIM to decide how relevant the response is w.r.t the context while training the generative model.

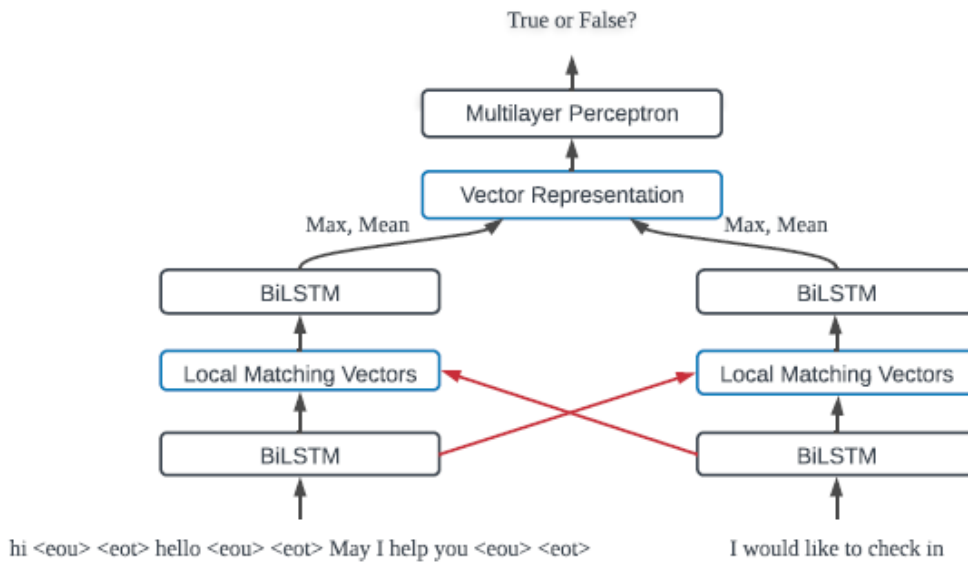


Figure-5:- ESIM Architecture [4]

BERT

BERT, or Bidirectional Encoder Representations from Transformers, was proposed by Jacob Devlin, Ming-Wei Chang, Kenton Lee, and Kristina Toutanova in 2018 [5]. It's a transformer-based bidirectional model that was pre-trained on a large corpus that included the Toronto Book Corpus and Wikipedia using a combination of masked language modelling objective and next sentence prediction. The goal of Masked Language Model (MLM) training is to hide (mask) a word in a sentence and then have the model guess which word was hidden (masked) based on the context. The goal of Next Sentence Prediction training is to get the model to predict whether two provided sentences are connected in some way or the other. This pre-trained model is easily available and can be applied to a variety of downstream tasks. This is referred to as fine-tuning.

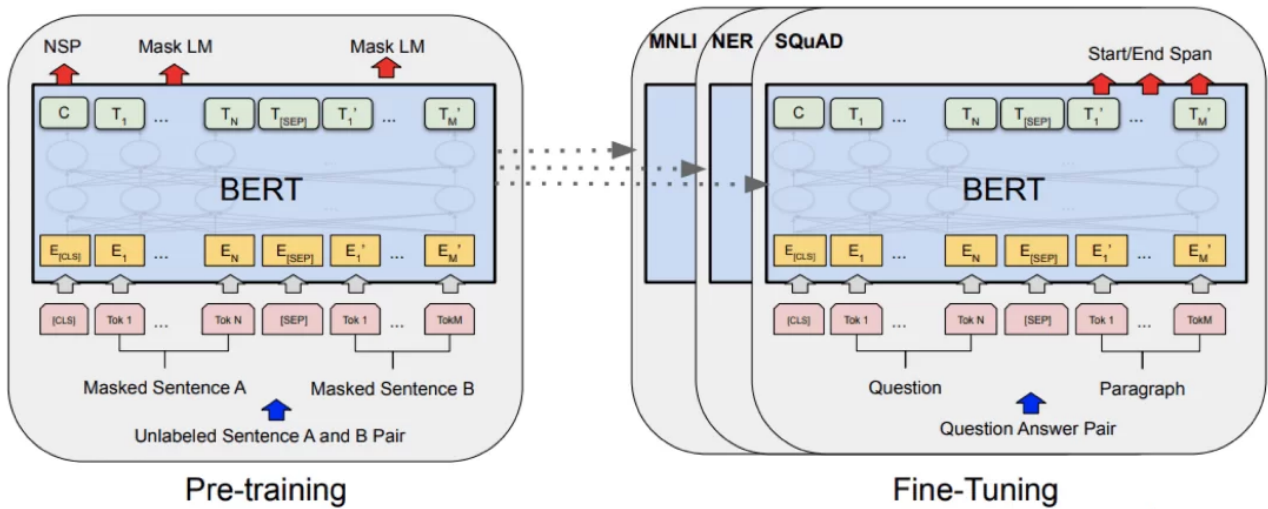


Figure 6:- Pre-training and Fine-Tuning of BERT [5]

We perform the following steps to do fine-tuning:-

- 1) We take the context and response, concatenate them and generate their embeddings using the BertTokenizer.
- 2) We pass them through the BERT model and concatenate the last four hidden layers
- 3) Then we pass them through a three layer Feed Forward neural network with ReLU activation function.
- 4) The above output is passed through a sigmoid function that gives us a probability score corresponding to the quality of the response given the context.
- 5) This is then used to calculate the cross-entropy loss which is then backpropagated to perform learning.

Just like ESIM, we use the class-1 probability from the trained BERT model to decide how relevant the response is w.r.t the context while training the generative model.

POLICY BASED REINFORCEMENT LEARNING

In policy-based learning, the model also known as the policy estimator tries to estimate the policy that best fits the scenario experienced by the model till now. The final output in this case is a probability distribution over the action space estimated using the policy learned by the model. The next action may then be chosen from the given probability distribution over actions based on a greedy or a sampling based approach.

REINFORCE [6] is a policy-based learning algorithm in which a policy estimator assigns probabilities to actions, the actions are chosen according to some rule (greedy or sampling), the chosen actions are tracked for a given batch and loss is accumulated as the sum of negative log probabilities of the chosen actions for a given batch multiplied by its reward. This loss is then back-propagated at the end of each batch.

As it might be evident that in our case, the Seq2Seq generative model acted as the policy-estimator network while the response generated by it was the action. In fact each token of the generated response was an action for which the model assigned probabilities and to which the reward was allocated. The learned retrieval model is analogous to the environment which outputs a reward corresponding to the action (generated response from Seq2Seq). The reward is then allocated to each token of the response based on the attention value given to that response token by the context, inside the attention layer used in ESIM/BERT. This ensures that the reward is allocated in a weighted manner with a greater reward allocated to good response tokens resulting in a smaller loss for those tokens. The “good response tokens” here are quantified based on how much attention is given to that token by all the context tokens, ie, higher the attention, more relevant is the token.

Algorithm 1 REINFORCE

Require: A differentiable policy parameterization $\pi(a \mid s, \theta)$

```
Initialize the parameters  $\theta$ 
Select step-size parameters  $0 < \alpha \leq 1$ 
Choose discount rate  $0 < \gamma \leq 1$ 
Choose max number of episodes  $N$ 
Choose number of episodes to batch together for an update  $K \geq 1$ 
1: while Episode  $n < N$  do:
2:   for  $K$  batches do:
     Generate an episode  $s_0, a_0, R_0, \dots, s_t, a_t, r_t$  following policy  $\pi(a \mid s, \theta)$ 
3:   for each step in the episode ( $t$ ), discount rewards do:
      $G_t \leftarrow \sum_{t=1}^T \gamma^t R_t$ 
4:   end for
     Calculate policy loss for all episodes in the batch  $\mathcal{L}(\theta) = -\frac{1}{m} \sum_t \ln(G_t \pi(a_t \mid s_t, \theta))$ 
     Update the policy:  $\theta \leftarrow \theta + \alpha \nabla \mathcal{L}(\theta)$ 
     Increment episode counter  $n \leftarrow n + 1$ 
5: end for
6: end while
```

EXPERIMENTS

DATASET

We use the **Daily Dialog** [7] dataset for our experiments. The dialogues in the dataset reflect our daily communication and cover various topics about our daily life. The language is human-written and less noisy. The dataset consists of 13,118 dialogues divided across training, validation and testing parts. Both the validation and testing components consist of 1000 dialogues and the remaining 11,118 dialogues are used for training purposes.

Total Dialogues	13,118
Average number of utterances per dialog	7.9
Average tokens per Dialogue	114.7
Average tokens per Utterance	14.6

Basic statistics of the Daily Dialog dataset

TRAINING

For training the retrieval models i.e BERT and ESIM we need to pre-process the dataset slightly. To do so we sample 10 random responses along with the ground truth response. These random responses serve as class 0 examples in the training dataset in addition to the ground truth which has a label 1. Rest of the steps used for training are the same as described in the Theory Section.

The conventional cross-entropy based methods for training generative models would rely on the actual target label that would be 1 if the response is appropriate for the given context and 0 otherwise. However the method we use for training relies on the scores obtained from the retrieval dialogue systems.

The various model specifics are shown in the table given below

Dimension of Hidden layer	1024
Number of encoder layers	4
Number of decoder layers	4
Number of heads in encoder	8
Number of heads in decoder	8
Dimension of the FF encoder layer	1024
Dimension of the FF decoder layer	1024
Maximum context length	300
Maximum response length	30

Model parameters

The generative model was trained using the **teacher forcing** [9] paradigm. That is the golden response is supplied to the decoder as an input and that is used to generate the output. Instead in a non-teacher forcing setup the output at the current time step is used as an input at the next time step. Some important parameters of the model are as follows:-

- 1) **Golden response ratio**:- The percentage of time the golden response is equal to the actual true response is given by the *grr* parameter. Let's say the *grr* is 0.9, then the ground truth response is supplied 90% of the times while 10% of the times an incorrect response is given.
- 2) **Margin**:- In addition we used a *margin* parameter to keep a limit over the reward i.e., margin was subtracted from the reward during training.

The results shown later are calculated using a *grr* of 0.9 and a *margin* of 0. We also use a linearly decaying schedule for the learning rate after a warm up period of 4 epochs. (except for the baseline transformer trained using cross entropy)

VALIDATION

We evaluate the performance of the retrieval model after each epoch and save the model having the minimum validation cross entropy loss.

Similarly we also evaluate the performance of the Seq2Seq model after each epoch on the validation dataset to control the overfitting and save the best model checkpoint. To perform validation we generate the response at each time step using the output at the last time step for each context in the validation dataset (non teacher forcing setup). The responses can be sampled using different techniques which are explained below. Once we obtain the response, both the context and response are passed to the retrieval model being used while training. The score obtained is used to save the best checkpoint i.e the checkpoint corresponding to the highest ESIM/BERT score is saved. The different decoding techniques are as follows:-

- 1) **Greedy decoding**:- In this strategy we greedily choose the tokens having the maximum probabilities at each time step to generate the output response
- 2) **Random sampling**:- As the name suggests here we randomly sample the tokens based on the probabilities assigned to each token by the model at every time step
- 3) **Top p or Nucleus sampling**:- The top t tokens having a cumulative probability exceeding p are chosen and the probabilities are re-distributed amongst them. As a result This method avoids choosing tokens having extremely low probabilities.

INFERENCE

For evaluating the performance of the retrieval models we use the **Recall@K** metric for 3 different values of K viz. 1, 10 and 50. This metric gives us the percentage of times the correct response is present among the top K responses returned by the retrieval model for a given context

For evaluating the performance of generative models we use the testing component of the Daily Dialog dataset. We generate the responses at each time step using the output at the previous time step and random sampling technique. These responses are then evaluated using several methods and compared with the baseline results obtained from the Seq2Seq model trained using Cross Entropy Loss. The various metrics used are as follows:-

- 1) **Retrieval model scores**:- The score assigned by both the retrieval models ESIM and BERT to the response generated by the Seq2Seq model subject to the context is referred to as the ESIM and BERT Score respectively.
- 2) **BLEU Score**:- The BLEU or Bilingual Evaluation Understudy Score [\[10\]](#) is a metric for evaluating the quality of the generated response as compared to a target reference.
- 3) **METEOR Score**:- The METEOR or Metric for Evaluation of Translation with Explicit Ordering Score [\[11\]](#) is another metric for understanding the quality of generated response with respect to a target output.
- 4) **LDA Topical Similarity**:- In this method of evaluation, we first train an LDA model [\[12\]](#) over the whole training corpus, taking each conversation, ground-truth response pair as one document i.e, we had 100K documents over which the LDA model was trained for a total of 100 topics. During inference, a topic probability vector was constructed for the context as well as the response and the similarity between them was computed using the Bhattacharyya Coefficient.

- 5) **Diversity Scores:-** One way to estimate the goodness of the generated response for a dialog model is to use the diversity metric [\[13\]](#). We measure the diversity using Distinct-1 and Distinct-2 metrics which are the ratio of distinct unigrams and bigrams respectively to the total number of tokens in all the generated responses together.
- 6) **Embedding Similarity Scores:-** Another method of evaluation was to find the similarity between sentence level embeddings for the generated and ground truth responses [\[14\]](#). A sentence level embedding may be constructed from the embedding vectors of its individual words based on different heuristics. In our case, we have used 3 different heuristics:-
- a) **Greedy Matching:** The overall similarity between the generated response r and the ground truth response r' is found by greedily matching the embedding vectors of the generated response tokens with those of the ground truth tokens. Similarly the reverse is done for the ground truth tokens and the final similarity is taken as the average of both similarities.
 - b) **Embedding Extrema:** The sentence level embedding is made by taking the extreme values along the respective embedding dimensions across all tokens of that sentence.
 - c) **Embedding Average:** The sentence level embedding is simply taken to be the average of the embeddings of its constituent response tokens

EVALUATION AND RESULTS

RETRIEVAL MODELS

In the tables shown below we summarise the results for different retrieval models.

	ESIM Model	BERT Model
Recall@1	45.42%	59.24%
Recall@10	81.30%	93.27%
Recall@50	98.06%	99.85%

Performance of retrieval models

We can clearly see that the pre-trained BERT retrieval model performs better than the ESIM retrieval model

GENERATIVE MODELS

In the table shown below we present the results for both the Seq2Seq models (trained using different retrieval models) and compare them to the baseline performance of Seq2Seq transformers trained using Cross Entropy loss (along with the Ground Truth).

<u>METRIC</u>	<u>Ground Truth</u>	<u>Seq2Seq with Cross Entropy</u>	<u>Seq2Seq with ESIM Retrieval Model</u>	<u>Seq2Seq with BERT Retrieval Model</u>
ESIM Score	0.3522	0.2046	0.2885	0.1596
BERT Score	0.8737	0.5234	0.6627	0.641
BLEU Score	-	0.0814	0.2068	0.1462
METEOR Score	-	0.0512	0.1831	0.1404
LDA Topical Similarity Score	0.168	0.1215	0.1503	0.1374
Distinct-1 Score	0.0844	0.0805	0.0675	0.0592
Distinct-2 Score	0.4292	0.3871	0.324	0.3379
Greedy Matching	-	0.6326	0.6944	0.6697
Embedding Extrema	-	0.3931	0.4931	0.4533
Embedding average	-	0.8176	0.8493	0.835

Performance of generative models

We can clearly see from the above results that the Seq2Seq models trained with the help of reinforcement learning using the scores obtained from retrieval models outperform the Seq2Seq models trained using cross entropy loss across most of the metrics. Among the two variants we observe that the ESIM based model performs better than the BERT based model in all the departments.

ABLATION STUDY

We first observe the effect of using top-p sampling instead of random sampling during inference time.

<u>METRIC</u>	<u>Ground Truth</u>	<u>Seq2Seq with Cross Entropy</u>	<u>Seq2Seq with ESIM Retrieval Model</u>	<u>Seq2Seq with BERT Retrieval Model</u>
ESIM Score	0.3522	0.1389	0.2755	0.1496
BERT Score	0.8737	0.433	0.6513	0.6144
BLEU Score	-	0.0637	0.2199	0.1528
METEOR Score	-	0.0407	0.199	0.1532
LDA Topical Similarity Score	0.168	0.1219	0.1511	0.1367
Distinct-1 Score	0.0844	0.169	0.0832	0.0691
Distinct-2 Score	0.4292	0.4731	0.3348	0.345
Greedy Matching	-	0.6127	0.698	0.6694
Embedding Extrema	-	0.3691	0.4997	0.4515
Embedding average	-	0.7692	0.8456	0.8298

Effect of using top-p sampling during inference

We now discuss some of the results obtained by varying the training conditions. Model M1 analyses the effect of not using the linear schedule with warm up for the learning rate for the Seq2Seq model with BERT as the retrieval model. Model M2 analyses the effect of using Xavier initialisation instead of the default initialisation for the Seq2Seq model with ESIM as the retrieval model

	Model M1	Model M2
ESIM Score	0.1613	0.1763
BERT Score	0.6273	0.4953
BLEU Score	0.0878	0.0766
METEOR Score	0.0647	0.0494
LDA Topical Similarity Score	0.1488	0.105
Distinct-1 Score	0.0634	0.0761
Distinct-2 Score	0.3636	0.3629
Greedy Matching	0.6374	0.6283
Embedding Extrema	0.4069	0.3982
Embedding average	0.8138	0.8095

Effect of varying the learning rate schedule and initialisation

The decline in the performance by not using any LR Schedule or using Xavier initialization is pretty evident from the above results.

Up until now we were choosing the best model as the one that gave the best reward during validation while generating the responses using previous outputs. This procedure was followed to mimic the testing environment as we won't be having the golden response during inference. However instead if we chose to mimic the training environment during validation i.e if we used teacher forcing and tried minimizing the validation loss then the following results are obtained. All the results shown below use Xavier initialization and don't use any learning rate schedule.

<u>METRIC</u>	<u>Seq2Seq with ESIM retrieval model</u>	<u>Seq2Seq with BERT retrieval model</u>
ESIM Score	0.1605	0.0328
BLEU Score	0.0806	0.0457
METEOR Score	0.0515	0.0276
LDA Topical Similarity Score	0.1054	0.0668
Distinct-1 Score	0.0783	0.1097
Distinct-2 Score	0.3693	0.5018
Greedy Matching	0.6348	0.596
Embedding Extrema	0.3991	0.3462
Embedding average	0.8182	0.7775

Effect of using teacher forcing during validation

Next we observe the effect of varying the *grr* and *margin* parameters. Model M1 analyses the impact of reducing the *grr* to 0.7 for the Seq2Seq model with BERT as the retrieval model. Model M2 analyses the effect of increasing the *margin* to 0.1 for the Seq2Seq model with ESIM as the retrieval model

	Model M1	Model M2
ESIM Score	0.0259	0.165
BLEU Score	0.0525	0.0793
METEOR Score	0.0318	0.0522
LDA Topical Similarity Score	0.0657	0.1014
Distinct-1 Score	0.0849	0.0735
Distinct-2 Score	0.4373	0.3434
Greedy Matching	0.6053	0.6389
Embedding Extrema	0.3557	0.4015
Embedding average	0.7044	0.8222

Effect of varying *grr* and *margin* parameters

CHAPTER-2: DMI BASED EVALUATION OF DIALOGUE SYSTEMS

LITERATURE REVIEW

Traditional dialogue evaluation metrics such as BERT, METEOR, ROUGE rely on the assumption that valid responses have significant word overlap with the ground truth responses. This assumption is clearly incorrect as there can be several possible responses for the same context. This is clearly illustrated in the figure shown below where there is no overlap between the model response and the ground truth response.

Context of Conversation
Speaker A: Hey John, what do you want to do tonight?
Speaker B: Why don't we go see a movie?

Ground-Truth Response
Nah, I hate that stuff, let's do something active.
Model Response
Oh sure! Heard the film about Turing is out!

Figure 7:- Why word overlap metrics are not appropriate for evaluation [14]

Another option is to use word embedding similarities to compare the ground truth response and the model's response as shown in [14]. Despite the fact that these metrics have been employed widely in literature for evaluating dialogue models, they reveal either a weak or no correlation with human judgments.

There are two types of dialogue evaluation metrics: referenced metrics, which compare the generated response to a specified ground-truth response and unreferenced metrics, which evaluate the generated response without making any such comparison. In an online situation, it is not possible to evaluate dialogue models using referenced metrics.

InferSent [15] is a sentence embeddings method that provides semantic sentence representations. It is trained on natural language inference data and generalizes well to many different tasks. The BERT-NLI model is similar to the InferSent model, except it replaces the LSTM encoder with a pre-trained BERT encoder. Tao et al. (2017) [16] presented the RUBER metric, which is a hybrid referenced-unreferenced metric that is trained without the need for human responses by bootstrapping negative samples straight from the dataset. MAUDE [17] is an unreferenced metric that employs state-of-the-art pretrained language models, along with a unique discourse structure aware text encoder, and a contrastive training technique that requires the model to distinguish between correct and randomly sampled negative responses.

THEORY

DMI SCORE

Unfortunately, existing language modelling (causal or masked) pretraining objectives are ineffective for modelling dialogues because the model is not directly trained to learn the content discourse structure, and such models are trained to generate responses word-by-word rather than predict a larger unit. Furthermore, because dialogue generation is one-to-many, the encoding model must be able to capture the uncertainty in the response prediction task, which most models do not.

To capture the intricate features of dialogues, we utilise a model that is pretrained using the Discourse Mutual Information (DMI) [18] based loss function and is known to outperform strong baselines by large margins in various dialogue understanding tasks. By using this model for evaluation we are able to capture a lot more complicated relationships between the context and response enabling us to get a metric that has significant alignment with the human judgements.

DISCOURSE MUTUAL INFORMATION BASED TRAINING OF MODEL

Mutual information between two random variables is defined as the reduction in the entropy of one random variable by knowing the value of the second random variable. The mutual information between two random variables representing two different segments inside the same discourse is defined as discourse mutual information (DMI). Let C and R refer to the context and response random variables. The goal is to learn continuous representations for C and R that can be used to estimate the true mutual information between them.

Let E_c and E_r be the representations for C and R based on some encoder. From Information Theory we know the following inequality.

$$I(C; R) \geq I(E_c; E_r)$$

For continuous-valued random variables, accurate computation of MI is not possible. Various variational lower bounds for calculating MI between continuous-valued random variables have been developed in recent years. When the MI estimator is included, the relationship looks as shown below

$$I(C; R) \geq I(E_c; E_r) \geq \hat{I}_{\theta}(E_c; E_r)$$

The loss function used to train the model is shown below.

$$\min_{\theta, \phi} \left[L_{\theta, \phi}(C, R) = -\hat{I}_{\theta, estimator}(E_c^{(\phi)}, E_r^{(\phi)}) \right]$$

Here θ denotes the parameters of the MI estimator and ϕ denotes the parameters of the encoder. The MI estimator used for the pre-trained model is InfoNCE-S [20] which is shown below.

$$I(C; R) \geq \log N - L_N$$

$$L_N = -\frac{1}{2} \mathbb{E}_{P_{CR}} \left[\log \frac{e^{f(c,r)}}{\sum_{r' \in R} e^{f(c,r')}} \right]$$

Here N is the batch size, $f(c,r)$ is the scoring function for a given context-response pair and c' , r' denote negative samples for context and response respectively.

ARCHITECTURE

The exact model architecture is shown in figure below.

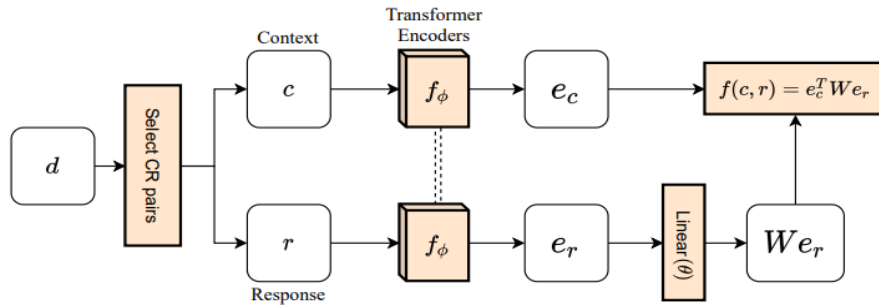


Figure 8:- Model architecture for DMI-Score metric [18]

Both the context as well as the response encoders are vanilla transformer models as proposed by Vaswani et al. The inputs begin with a [CLS] token. The different context utterances are separated by [EOU] tokens. The maximum number of tokens in the context is limited to 300 while this limit is set to 60 for the responses. After encoding by the transformers the response is passed through a linear neural network layer (W). Suppose the encoded outputs are e_c and e_r for the context and response respectively, the final score is given by the dot product of e_c and We_r .

PROPOSED EVALUATION METRIC

In order to use DMI as an evaluation metric we make use of the above model that is pre-trained using the DMI based loss function. Given a new context-response pair we pass the context and response through their encoders and linear layers (for response) and calculating the score as $e_c^T We_r$.

But since that score alone is insufficient to describe the quality of response given the context we additionally generate another P utterances randomly sampled from the Daily Dialog dataset and fix this response pool. Then we pair each of these responses with the original context and score a total of $P+1$ C-R pairs for the same context. We sort these scores in ascending order and if the position of the actual response score is pos then the final score is given by $pos/(P+1)$.

By default we use a P value of 1000 in our work.

EXPERIMENTS

BASELINES

We use four baseline evaluation metrics in our experimentation. They are described below

InferSent

The authors of InferSent proposed to learn universal sentence representations using the Stanford Natural Language Inference dataset. For encoding the sentences they make use of a bi-directional LSTM. For evaluation of a given context response pair we concatenate the encoded context and response along with their element wise product and difference. The output is then passed through a feed forward neural network and the sigmoid activation function is used at the output to get the desired score.

DistilBERT-NLI

This is exactly identical to InferSent except for the fact that the encoder used is a pre-trained DistilBERT [\[21\]](#) model instead of a bi-directional LSTM.

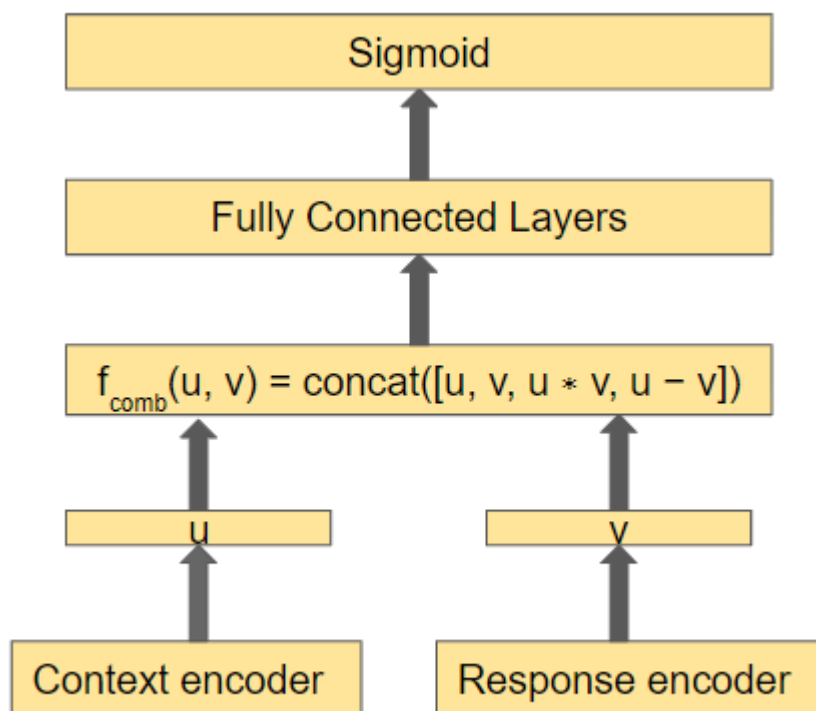


Figure 9:- Model architecture for InferSent and DistilBERT-NLI

RUBER

RUBER stands for “Referenced metric and Unreferenced metric Blended Evaluation Routiner”. It consists of a referenced metric that measures the similarity between the ground truth and the generated response by pooling of word embeddings along with an unreferenced metric that we use in our baselines. A Bi-GRU RNN is used to score the context and generated response along with a multi-layered perceptron (MLP) in the unreferenced metric.

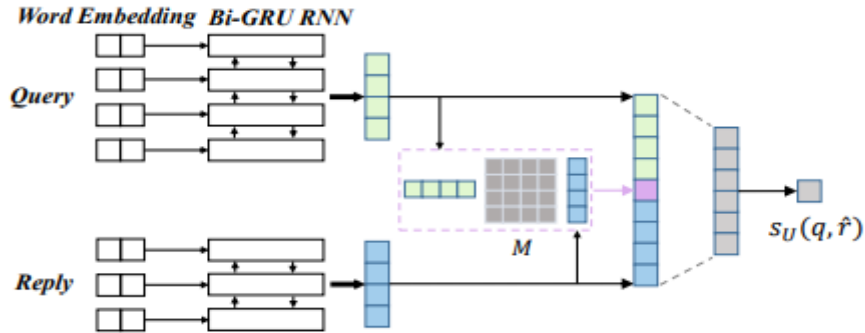


Figure 10:- Model architecture for RUBER unreferenced metric [17]

MAUDE

MAUDE also follows an approach similar to InferSent to score a given context-response pair but uses Noise Contrastive Estimation [22] for training. Specifically the model is trained to differentiate between a correct response and an incorrect response generated by varying the syntax or semantics. Moreover they use a specialised BERT based encoder and additionally model dialogue transitions using a recurrent neural network.

CORRELATION WITH HUMAN JUDGEMENTS

We evaluate the metrics by measuring their correlation with the human judgements on 18265 conversations. See et al [19] conducted a large-scale human evaluation of 28 model configurations to study the effect of controllable attributes in dialogue generation. They publicly released the chatlogs of several human-model and human-human conversations along with the associated human judgements based on a Likert scale on eight parameters viz. fluency, engagingness, humanness, making sense, inquisitiveness, interestingness, avoiding repetition and listening. We use the chatlogs to get scores from our evaluation metrics and measure the Spearman’s correlation with the human judgements. The original authors proposed to use a multi-step evaluation methodology where they rated an entire dialog and not individual context response pairs. So we average the scores for different turns to get an aggregate score for the entire dialog. Lastly we also calculate the mean of eight correlations to get an idea of the overall similarity with human judgements.

Algorithm 1 Evaluation using DMI-Score

Input Context C , response R , pretrained DMI model consisting of transformer encoders f_ϕ and weight matrix W , response pool RP of size P

```
1: initialize  $SC = []$  ▷ empty array
2: Prepend  $R$  to  $RP$ 
3:  $e_c = f_\phi(C)$ ,  $e_{resp} = f_\phi(R)$  ▷ encoded context and response vector
4: for  $t = 0$  to  $P$  do
5:    $e_r = f_\phi(RP[t])$ 
6:    $f(c, r) = e_c^T W e_r$  ▷ calculate score
7:   append  $[f(c, r), SC]$  ▷ append to score array
8: end for
9: sort  $(SC, asc)$  ▷ sort array in ascending order
10:  $pos = index[e_c^T W e_{resp}, SC]$  ▷ get index of original response
```

Output $pos/(P + 1)$

EVALUATION AND RESULTS

	RUBER	InferSent	DistilBERT-NLI	MAUDE	DMI-Score
Fluency	0.322	0.246	0.443	0.37	0.059
Engagingness	0.204	0.091	0.192	0.232	0.416
Humanness	0.057	-0.108	0.129	0.095	0.367
Making Sense	0.0	0.005	0.256	0.208	0.238
Inquisitiveness	0.583	0.589	0.598	0.728	0.041
Interestingness	0.275	0.119	0.135	0.24	0.443
Avoiding Repetition	0.093	-0.118	-0.039	-0.035	0.14
Listening	0.061	-0.086	0.124	0.112	0.471
Mean	0.199	0.092	0.23	0.244	0.272

Performance of DMI-Score evaluation metric along with the baselines

We can clearly see that the DMI-Score outperforms all the other metrics in five out of the eight parameters as well as the mean correlation score.

CONCLUSION AND FUTURE WORK

We can clearly see the improvement in the performance across all the metrics for the retrieval based Seq2Seq models as compared to the baseline models. Moreover upon analysing the response quality we observe a clear distinction between the two. The primary reason behind the improvement in performance is the usage of reinforcement learning to distribute the reward across different tokens using the attention weights obtained from the trained retrieval model and the ability to provide goodness scores for a varied set of responses irrespective of the ground truth during training. The next step would be to run this experiment for an even larger dataset such as the Reddit dataset. This would bring about a further improvement in the performance. Moreover we could extend this work to other different dialogue generation tasks and compare the results with state of the art baselines.

We also see a clear improvement in the DMI-Score evaluation metric as compared to other baselines. This can be attributed to the novel discourse mutual information based loss function used while training the model which enables it to understand and capture the relationships between different utterances of a dialogue. The next step would be to measure the robustness of this metric by calculating the difference between the scores assigned to the ground truth and artificially generated positive/negative responses upon varying the syntax/semantics of the ground truth for a given context.

REFERENCES

- [1] Ashish Vaswani et al. “Attention is All you Need”. In: *Advances in Neural Information Processing Systems*. Ed. by I. Guyon et al. Vol. 30. Curran Associates, Inc., 2017.
- [2] D Bahdanau et al. “Neural machine translation by jointly learning to align and translate”. In *CoRR* abs/1409.0473 (2014)
- [3] Qian Chen et al. “Enhancing and Combining Sequential and Tree LSTM for Natural Language Inference”. In: *CoRR* abs/1609.06038 (2016)
- [4] Qian Chen and Wen Wang. “Sequential Attention-based Network for Noetic End-to-End Response Selection”. In: *CoRR* abs/1901.02609 (2019)
- [5] Devlin, J.; Chang, M.-W.; Lee, K.; and Toutanova, K. 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)*, 4171–4186. Minneapolis, Minnesota: Association for Computational Linguistics.
- [6] Williams, R.J. Simple statistical gradient-following algorithms for connectionist reinforcement learning. *Mach Learn* 8, 229–256 (1992). <https://doi.org/10.1007/BF00992696>
- [7] Yanran Li et al. “DailyDialog: A Manually Labelled Multi-turn Dialogue Dataset”. In: *Proceedings of the Eighth International Joint Conference on Natural Language Processing (Volume 1: Long Papers)*. Taipei, Taiwan: Asian Federation of Natural Language Processing, Nov. 2017, pp. 986–995.
- [8] Matthew E. Peters et al. “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)*. New Orleans, Louisiana: Association for Computational Linguistics, June 2018, pp. 2227–2237.
- [9] Samy Bengio et al. “Scheduled Sampling for Sequence Prediction with Recurrent Neural Networks”. In: *CoRR* abs/1506.03099 (2015)
- [10] Kishore Papineni et al. “Bleu: a Method for Automatic Evaluation of Machine Translation”. In: *Proceedings of the 40th Annual Meeting of the Association for Computational Linguistics*. Philadelphia, Pennsylvania, USA: Association for Computational Linguistics, July 2002, pp. 311–318.
- [11] Satanjeev Banerjee and Alon Lavie. 2005. METEOR: An Automatic Metric for MT Evaluation with Improved Correlation with Human Judgments. In *Proceedings of the ACL Workshop on Intrinsic and Extrinsic Evaluation Measures for Machine Translation and/or Summarization*, pages 65–72, Ann Arbor, Michigan. Association for Computational Linguistics.
- [12] David M. Blei et al. “Latent dirichlet allocation”. In: *Journal of Machine Learning Research* 3 (2003), p. 2003.

- [13] Jiwei Li et al. “A Diversity-Promoting Objective Function for Neural Conversation Models”. In: *Proceedings of the 2016 Conference of the North American Chapter of the Association for Computational Linguistics: Human Language Technologies*. San Diego, California: Association for Computational Linguistics, June 2016, pp. 110–119.
- [14] Chia-Wei Liu et al. “How NOT To Evaluate Your Dialogue System: An Empirical Study of Unsupervised Evaluation Metrics for Dialogue Response Generation”. In: Mar. 2016, p. 13. doi: 10.18653/v1/D16-1230.
- [15] Alexis Conneau et al. “Supervised Learning of Universal Sentence Representations from Natural Language Inference Data”. In: *Proceedings of the 2017 Conference on Empirical Methods in Natural Language Processing. Copenhagen, Denmark: Association for Computational Linguistics*, Sept. 2017, pp. 670–680.
- [16] Chongyang Tao et al. “RUBER: An Unsupervised Method for Automatic Evaluation of Open-Domain Dialog Systems”. In: *CoRR* abs/1701.03079 (2017)
- [17] Koustuv Sinha et al. “Learning an Unreferenced Metric for Online Dialogue Evaluation”. In: *Proceedings of the 58th Annual Meeting of the Association for Computational Linguistics. Online: Association for Computational Linguistics*, July 2020, pp. 2430–2441.
- [18] Bishal Santra et al. “Representation Learning for Conversational Data using Discourse Mutual Information Maximization”. In: *CoRR* abs/2112.05787 (2021)
- [19] Abigail See et al. “What makes a good conversation? How controllable attributes affect human judgments”. 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)*. Minneapolis, Minnesota: Association for Computational Linguistics, June 2019, pp. 1702–1723.
- [20] Aäron van den Oord, Yazhe Li, and Oriol Vinyals. “Representation Learning with Contrastive Predictive Coding”. In: *CoRR* abs/1807.03748 (2018).
- [21] Victor Sanh et al. “DistilBERT, a distilled version of BERT: smaller, faster, cheaper and lighter”. In: *5th Workshop on Energy Efficient Machine Learning and Cognitive Computing @ NeurIPS 2019*. 2019.
- [22] Michael Gutmann and Aapo Hyvarinen. 2010. Noise-contrastive estimation: A new estimation principle for unnormalized statistical models. In *Proceedings of the Thirteenth International Conference on Artificial Intelligence and Statistics*.