Asking Clarifying Questions

Ananya Parikh, Neha Chowdhary, Kumar Pranjal Tripathi, Rashi Goyal

1 Introduction

The process of Natural Language Processing (NLP) involves analyzing and understanding human language to enable computers to interact with humans in a more natural way. This report explores the importance of asking clarifying questions in NLP, the various types of clarifying questions and strategies for asking meaningful clarifying questions. It also discusses how to use clarifying questions to improve accuracy and reduce ambiguity in NLP applications and provides examples of how clarifying questions are used in various NLP tasks like information retrieval and text classification. The ultimate goal of a conversational system is to assist users by returning an appropriate answer in response to their requests. The task of 'Clarifying Questions' involves development of an open domain dialogue system in which the user asks a question to the system and in return the system analyze that the asked question need a clarifying question to be asked, i.e. whether the question contain many possible answers in the database and then the system decides to ask a clarifying question in return to meaningfully answer initially asked question. On the other part if the asked question is clear and has no ambiguity with multiple answers then the system returns the answer corresponding to the asked query.

2 Methods and Dataset

The task of asking clarifying question is divided into to sub task: classification and retrieval. Classification involve the user asking a question in form of a query q and the model needs to predict that whether the query needs a clarifying question to be asked or not. Retrieval model return the answer to the question asked based on the information that whether the given query need a clarifying question or not. Our main goal is to develop a open domain dialogue system that respond to a user query and identify whether a clarifying question need to be asked or not, if a clarifying question is not required then the answer corresponding to the question will be prompted back to the user in the course of question answering.

For the task, we are using the **ClariQ** dataset with a size of 4000 thousand for the *question bank* and 2300 for the *dev* test(training set). The data sets and the scripts for automatic evaluation can be found at the ClariQ repository and also in our project repository.

2.1 Classification Models

BERT base, DistilBERT, RoBERTa, GLOVE and LSTM models have been

implemented for the classification task. The classification task has four classes to classify the task on a discrete scale of *1-4* being scores signifying the need for clarification. **Bidirectional Encoder Representation for Transformer (BERT)**, an NLP model developed by Google Research in 2018, is just an encoder stack of transformer architecture. For our text classification task, we use *bert-base-uncased* having 12 layers as our pre-trained model by fine tuning it by freezing all the layers of the model and appending a dense and a *softmax* layer to it. **DistilBERT** model is a distilled version of BERT released by Google which leveraged knowledge distillation during the pre-training phase and showed that it is possible to reduce the size of a BERT model by 40% while retaining 97% of its language understanding capabilities and being 60% faster. One important point to be noted that DistilBERT accepts a *max_sequence_length* of 512 tokens.

RoBERTa model shares the same architecture as the BERT model excpet that it doesn't use the next-sentence pre-training objective like BERT but is trained with much larger mini-batches and learning rates. In addition, RoBERTa uses a different pre-training scheme and substitutes a character-level BPE vocabulary for a byte-level BPE tokenizer (similar to GPT-2). The GloVe (Global Vectors for Word Representation) is an unsupervised learning algorithm that is trained on a big corpus of data to capture the meaning of the words by generating GloVe Embeddings. We have used the 'glove.42B.300d' pretrained glove word vector using Keras. LSTM (Long-Short Term Memory) is a type of recurrent neural network in which we use a multiple word string to find out the class to which the text belongs using appropriate layers of embedding and encoding.

2.2 Retrieval Models

The retrieval model implemented are **Bm25okapi, XLNet, ERNIE-ranker and ELECTRA**. **BM25Okapi** model is a widely used ranking algorithm for information retrieval tasks and is based on the Okapi BM25 algorithm. To use the BM25Okapi model for the retrieval task of the clarifying question project of NLP, we define a corpus of questions and answers that you want to retrieve from, pre-process the text by removing stop words, stemming, and converting the text to lowercase. Then we tokenize the text into terms and calculate the BM25 score for each query and document pair using the BM25Okapi formula and rank the documents based on the BM25 score and return the top-k documents as potential clarifying questions.

XLNet is a state-of-the-art pre-trained language model that can be used as a retrieval model to identify potential clarifying questions. One method is to use the dot product of the encoded input question and potential clarifying question as the similarity score between the encoded input question and a potential clarifying question. This approach is often referred to as *semantic matching* or *semantic similarity*.

ERNIE-Ranker is a pre-trained language model that is based on the ERNIE (Enhanced Representation through knowledge Integration) architecture, which integrates knowledge graphs and external knowledge sources into the pre-training process. ERNIE-Ranker is fine-tuned on the task of relevance ranking, which involves predicting the relevance of a candidate answer to a given question.**ELECTRA** (Efficiently Learning an Encoder that Classifies Token Replacements Accurately) model's pre-training process is based on the idea of replacing input tokens with generated tokens, and then training a classifier to distinguish between the original and generated tokens.

3 Result and Analysis

3.1 Classification Results

3.1.1 BERT and DistilBERT

	precision	recall	f1-score	support
0	0.00	0.00	0.00	61
1	0.51	0.97	0.67	314
2	0.36	0.03	0.05	199
3	0.00	0.00	0.00	34
accuracy			0.51	608
macro avg	0.22	0.25	0.18	608
weighted avg	0.38	0.51	0.36	608

(a) Figure 1.1

```
Epoch 1/3
455/455 [==============================] - 58s 73ms/step - loss: 0.1535 - accuracy: 0.9452 - val_loss: 0.0015 - val_accuracy:
1.0000
Epoch 2/3
455/455 [==============================] - 26s 56ms/step - loss: 0.0011 - accuracy: 1.0000 - val_loss: 3.9135e-04 - val_accura
cy: 1.0000
Epoch 3/3
455/455 [==============================] - 25s 55ms/step - loss: 3.9219e-04 - accuracy: 1.0000 - val_loss: 1.7346e-04 - val_ac
curacy: 1.0000
: <keras.callbacks.History at 0x7f35d57e4e50>
```

(b) Figure 1.2

Figure 1: BERT (Fig. 1.1) and DistilBERT (Fig. 1.2) results

3.1.2 RoBERTa, GLOVE and LSTM

```
101/101 [==============================] - 175s 784ms/step - loss: 10.5391 - accuracy: 0.3604 - val_loss: 1.3795 - val_accurac
y: 0.3734
Epoch 2/3
101/101 [==============================] - 19s 189ms/step - loss: 1.3637 - accuracy: 0.3808 - val_loss: 1.3472 - val_accuracy:
0.3734
Epoch 3/3
101/101 [==============================] - 19s 190ms/step - loss: 1.3317 - accuracy: 0.3808 - val_loss: 1.3187 - val_accuracy:
0.3734
```

(a) Figure 1.2

GLOVE

```
Epoch 19/20
58/58 [==============================] - 27s 468ms/step - loss: 1.1961 - accuracy: 0.4275 - val_loss: 1.2522 - val_accuracy:
0.3804
Epoch 20/20
58/58 [==============================] - 28s 479ms/step - loss: 1.1945 - accuracy: 0.4288 - val_loss: 1.2506 - val_accuracy:
0.3837
```

LSTM

```
Epoch 1/30
102/102 [==============================] - ETA: 0s - loss: 1.1112e-06 - accuracy: 1.0000
The accuracy of the training set and the validation set has reached more than 90%!
102/102 [==============================] - 5s 45ms/step - loss: 1.1112e-06 - accuracy: 1.0000
```

(b) Figure 1.2

Figure 2: RoBERTa, GLOVE and LSTM results

3.2 Retrieval Results

3.2.1 Bm25okapi

Model	R@5	R@10	R@20	R@30	NDCG1	NDCG3	NDCG5	NDCG10	NDCG20
bm25plus	0.3222	0.5585	0.6705	0.6899	0.1859	0.1565	0.1521	0.1330	0.1246
bm25okapi	0.3245	0.5638	0.6675	0.6913	0.1859	0.1607	0.1530	0.1357	0.1281

(a) Figure 4.1

P1	P3	P5	P10	P20	MRR100
0.225	0.1833	0.1737	0.1356	0.1128	0.2980
0.2312	0.1896	0.1750	0.1394	0.1178	0.3090

(b) Figure 4.2

Figure 3: Bm25okapi

3.2.2 BERT-Ranker, BERT-reranker, Bm25 and ELECTRA

	value			
metric	Recall5	Recall10	Recall20	Recall30
model				
bm25	0.3246	0.5638	0.6675	0.6913
BERT-reranker	0.3487	0.6176	0.6913	0.6913
BERT-ranker	0.3523	0.6229	0.7290	0.7580

(a) Figure 5.1

	value										
metric	MRR100	NDCG1	NDCG10	NDCG20	NDCG3	NDCG5	P1	P10	P20	P3	P5
model											
bm25	0.3096	0.1859	0.1363	0.1285	0.1608	0.1530	0.2313	0.1406	0.1181	0.1896	0.1750
BERT-ranker	0.3232	0.1708	0.1560	0.1512	0.1674	0.1611	0.2188	0.1669	0.1400	0.1979	0.1825
BERT-reranker	0.3232	0.1708	0.1560	0.1512	0.1674	0.1611	0.2188	0.1669	0.1400	0.1979	0.1825

(b) Figure 5.2

	value			
metric	Recall5	Recall10	Recall20	Recall30
model				
Electra-ranker	0.0000	0.0013	0.0042	0.0059
Electra-reranker	0.1654	0.3057	0.5412	0.6913
bm25	0.3246	0.5638	0.6675	0.6913

(c) Figure 5.3

Figure 4: BERT-Ranker, BERT-reranker, Bm25 and ELECTRA

3.2.3 ERNIE

	value			
metric	Recall5	Recall10	Recall20	Recall30
model				
bm25	0.3246	0.5638	0.6675	0.6913
ERNIE-ranker	0.3311	0.5879	0.6925	0.7133
ERNIE-reranker	0.3340	0.5989	0.6849	0.6913

(a) Figure 6.1

	value										
metric	MRR100	NDCG1	NDCG10	NDCG20	NDCG3	NDCG5	P1	P10	P20	P3	P5
model											
bm25	0.3096	0.1859	0.1363	0.1285	0.1608	0.1530	0.2313	0.1406	0.1181	0.1896	0.1750
ERNIE-ranker	0.3225	0.1828	0.1533	0.1438	0.1733	0.1655	0.2313	0.1613	0.1303	0.2042	0.1925
ERNIE-reranker	0.3349	0.1953	0.1580	0.1474	0.1810	0.1720	0.2437	0.1644	0.1319	0.2104	0.1975

(b) Figure 6.2

Figure 5: ERNIE

4 Analysis of Results

Seven model related to classification and retrieval has been implemented and their corresponding accuracy have been compared. Overall, this report highlights the critical role of clarifying questions in NLP and provides valuable insights for practitioners seeking to improve their NLP skills. Among different classification model the **DistilBert** proved to be highly efficient on the present dataset and had earned accuracy of more than **97%**, **LSTM** also achieved an efficiency of more than **97%**, **RoBERTa and Glove** has an accuracy of around **40%**. Among the different Retrieval model the **Erine-Ranker** model proved to be the most efficient with the highest **MRR100** value, **Bm25** model gave considerable good results with an efficiency close to the **ERNIE** model.