Investigating Query Expansion and Coreference Resolution in Question Answering on BERT

Santanu Bhattacharjee[2], Rejwanul Haque[1,2][0000−0003−1680−0099], Gideon Maillette de Buy Wenniger[1,2], and Andy Way[1,2][0000−0001−5736−5930]

[1] ADAPT Centre,
[2] School of Computing, Dublin City University, Dublin, Ireland
santanu.bhattacharjee2@mail.dcu.ie
{rejwanul.haque,gideon.debuywenniger,andy.way}@adaptcentre.ie

Abstract. The Bidirectional Encoder Representations from Transformers (BERT) model produces state-of-the-art results in many question answering (QA) datasets, including the Stanford Question Answering Dataset (SQuAD). This paper presents a query expansion (QE) method that identifies good terms from input questions, extracts synonyms for the good terms using a widely-used language resource, WordNet, and selects the most relevant synonyms from the list of extracted synonyms. The paper also introduces a novel QE method that produces many alternative sequences for a given input question using same-language machine translation (MT). Furthermore, we use a coreference resolution (CR) technique to identify *anaphors* or *cataphors* in paragraphs and substitute them with the original referents. We found that the QA system with this simple CR technique significantly outperforms the BERT baseline in a QA task. We also found that our best-performing QA system is the one that applies these three preprocessing methods (two QE and CR methods) together to BERT, which produces an excellent F_1 score (89.8 F_1 points) in a QA task. Further, we present a comparative analysis on the performances of the BERT QA models taking a variety of criteria into account, and demonstrate our findings in the answer span prediction task.

Keywords: query expansion · coreference resolution · question answering · information retrieval · machine translation · neural machine translation.

1 Introduction

Text-based QA systems have proven to be a crucial technique for IR since users can obtain the information that they need while avoiding having to go through thousands of documents. As far as recent research in QA is concerned, attention-based neural network (NN) architectures [9, 5] have shown their potential in this task and produced promising results. ELMo [17], a character-based context-aware representation model, was shown to be useful at addressing this problem, while solving the out-of-vocabulary (OOV) problem by allowing the NN model to generate embeddings for the OOV words. Recently, Vaswani et al. [24] introduced Transformer as an efficient alternative to recurrent or convolutional NNs. The encoder-decoder architecture with attention mechanism has shown promising results on MT tasks. Based on the Transformer architecture, Delvin et al.

[5] proposed a powerful NN architecture – BERT – for a variety of NLP tasks including QA. BERT has significantly impacted many areas of natural language processing (NLP), e.g. QA has reached new heights on SQuAD [20]. BERT provides context-aware bidirectional representations from an unlabeled text by jointly conditioning from both the left and right contexts within a sentence, and can also be used as a pre-trained model with one additional output layer to fine-tune downstream NLP tasks, such as QA. Considering the recent success of BERT in QA, we have taken the BERT QA model as the baseline in our work.

Machine reasoning is at the core of solving a QA problem artificially, and it requires an understanding of natural language. Natural language understanding (NLU) is considered to be a complex task since it comes with its own challenges, such as word-sense disambiguation, existence of coreferencing entities, and understanding syntactic and semantic similarities between words. This work aims to address some of these problems by providing the learning model with more reasoning knowledge about enriching input questions or resolving references in paragraphs. CR [12] is regarded as a challenging task in many NLP tasks (e.g. MT), and has also been moderately investigated in QA [13, 25, 22]. In this work, we identify anaphors or cataphors (expressions referring to the same entity in a text passage) in paragraphs and substitute them with the original referring entities. The intuition underpinning this is that such preprocessing can provide the learning model more direct knowledge. For example, the pronoun 'He' refers to 'Sam' in the following paragraph "*Sam is moving to London. ... He has got a job there*"; replacing the pronominal entity 'He' with referent 'Sam' in the second sentence can add more direct knowledge to the QA model.

The semantic similarities between words are the other aspects of NLU, which were considered for investigation in this work. We present two novel QE techniques, the first one using a lexical knowledge base (WordNet [14]), and the second one using same-language MT [1]. Although the knowledge bases were heavily used for automatic QE [6, 3], this work presents a novel technique that identifies good terms from a given input question following a state-of-the-art term classification method, extracts synonyms of the good terms using WordNet, and selects the most relevant synonyms from the list of extracted synonyms. Same-language MT was successfully used in many NLP applications, e.g. text-to-speech synthesis for creating alternative target sequences [1], translation between varieties of the same language (Brazilian Portuguese to European Portuguese) [7], and paraphrase generation [18]. In this work, we developed an English-to-English MT system using the state-of-the-art Transformer model [24]. The MT system is able to generate n-best (same-language) translations for a given question, which can be viewed as the alternative sequences of the input question. These QE methods can considerably enrich the contexts of the input questions, and add extra reasoning knowledge to the QA model.

In this work, we carried out experiments applying these QE and CR techniques in QA individually and collaboratively taking the state-of-the-art BERT model into account. Rondeau and Hazen [21] analysed the outputs of a number of QA models applied to SQuAD to identify the core challenges for the QA systems on this data set. Since the introduction of BERT to the NLP community, researchers have been investigating the strength and weakness of BERT on the downstream tasks including QA [19, 23]. This

work also presents a comparative analysis on the ability of the baseline and our best-performing QA models to predict the answers correctly on SQuAD, taking a variety of criteria into account.

2 Baseline QA System

BERT, which makes use of the Transformer architecture, provides context-aware bidirectional representations from an unlabeled text by jointly conditioning from both the left and right contexts within a sentence. In short, BERT is made of a stack of encoders where each encoder consists of two sub-layers; the first sub-layer is a multi-head attention layer and the second sub-layer is a simple feed forward network. It can also be used as a pre-trained model with one additional output layer to fine-tune downstream NLP tasks, such as QA. For fine-tuning, the BERT model is first initialized with the pre-trained parameters, and all of the parameters are fine-tuned using the labeled data from the downstream tasks. Considering the recent success of BERT in QA, we have taken the BERT QA model as the baseline in our work. We used the SQuAD 1.1 dataset [20] to fine-tune the pre-trained BERT model. Given a paragraph from Wikipedia and a question relating to the paragraph, the task is to predict the answer text span in the paragraph. There are two architectural variations of the BERT model: $\text{BERT}_{\text{BASE}}$ and $\text{BERT}_{\text{LARGE}}$. These two architectures differ only in the size of the network layers and dimensions. In our experiments, we considered $\text{BERT}_{\text{BASE}}$ as our baseline.

3 Our Methods: Enriching Questions and Paragraphs

3.1 Query Expansion with WordNet

Query expansion is a commonly used method for mitigating the vocabulary mismatch problem in many NLP tasks. As far as QA is concerned, synonymous variations of an important word or phrase in a question need to be taken into account since variations instead of the actual word or phrase may appear in the paragraph that contains the answer. In theory, the word embedding layers of BERT should help address this to a certain extent. Additionally, we believe that injecting further context in the form of synonymous variations of the important words of the questions to a QA model would help it to find the right answers.

In this context, Cao et al. [2] showed that terms in a query can be categorized as good, bad and neutral. The good terms in a query help in finding the information from the text. Cao et al. [2] used features like term distribution, co-occurrence relations, weighted term proximity, and proposed a supervised learning method (SVM) for classifying the good, bad and neutral terms of a given query. In this work, first we identify those words of a question that are more important than others in getting the right answer from the paragraph, and then we further expand them in order to include more reasoning knowledge to the question. In other words, given a question, we identify its good terms and extract the most relevant synonyms of each of the good terms. We followed [2] and considered their features in order to build a classifier. In our case, we used a state-of-the-art classification algorithm: long short-term memory (LSTM) network [8]. We found that the

LSTM classifier performed competently in the classification task (predicting good, bad or neutral terms) (we obtained an F_1 score of 81.3 on a held-out test set).

As mentioned above, we considered good terms only in our query expansion process. First, we expand abbreviated good terms, if any, into full forms, e.g. *V.P.* is expanded to *Vice President*, *Dr.* is expanded to *Doctor*. For this, we used a python toolkit abbreviate (v 0.1.1).[1] WordNet was used to obtain the synsets for the good terms. However, for a given good term, we chose the most relevant synonyms from the synset. We measured cosine and semantic similarities between the good term and its synonyms. The term (A) and a synonym (B) are represented as distributed continuous vectors, which were obtained using the BERT pre-trained model. The cosine similarity is computed by taking the dot product of two vectors as shown in (1):

$$A \cdot B = ||A||||B|| \cos\theta \tag{1}$$

Semantic similarity between two words is measured using Wu-Palmer similarity [26]. The similarity score denotes how similar two word senses are, based on the depth of the two senses in WordNet. In order to measure the Wu-Palmer similarity between two words, we made use of the NLTK python toolkit.[2] A synonym for a good term is selected when the cosine and semantic similarity scores are above a threshold value. To exemplify, consider the question "*In what year did the CIA establish its first training facility?*" from SQuAD. The LSTM classifier identifies 'CIA', 'establish', 'training', and 'facility' as the good terms of the question. For each of the good terms we obtain a list of synonyms from WordNet, e.g. 'establish': 'set up', 'constitute', 'launch', 'plant', etc. Then, the most relevant synonyms (e.g. 'establish': 'set-up', 'launch') for each good term were identified following the startegy mentioned above. The resulting list of relevant synonyms for all good terms were then appended to the question. The expanded input question and the paragraph are represented as a single packed sequence.

3.2 Query Expansion with Neural MT

Translation of a source sentence into a target language can be generated in numerous ways. Similarly, in our case, a question can be represented in various forms. We developed a same-language MT system (English-to-English) that can generate n-best translations for an input sentence. In order to obtain different forms of a question, we translate it with the same-language MT system. The resulting n-best translations can be viewed as the alternative sequences of the question.

In our work, in order to build the MT system, we considered Transformer [24] which is regarded as the current state-of-the-art in MT research. We used the MarianNMT [10] toolkit and the European parliamentary proceedings (Europarl) corpus [11] for the NMT training. The training, development and test sets contains 13,201,483, 2,146 and 1,000 sentences, respectively. Additionally, we took high scoring five million English paraphrases from Multilingual Paraphrase Database[3] [16] and appended them to the training data. In our experiments, we followed the recommended best set-up by Vaswani et al.

[1] `https://pypi.org/project/abbreviate/`
[2] `http://www.nltk.org/`
[3] `http://paraphrase.org/#/download`

[24]. We obtained 99.69 BLEU [15] on the development set. The English-to-English NMT system was tested on a held-out test set, and we obtained 94.19 BLEU points on the test set. As you can see that the BLEU scores (on the development and test sets) are unusually high. This is because MT is being done on same-language (i.e. English-to-English). SQuAD includes 87,599 questions, which were translated with the English-to-English NMT system. Thus, we obtained alternative sequences for the questions.

The NMT-based QE process provides variants for a given input question, which are appended to the original question. The expanded input question and the paragraph are represented as a single packed sequence as in above (cf. Section 3.1). As mentioned above, the NMT system produced an *n*-best list for a given question. In this set-up, we experimented with different sizes of *n* (3, 5, 7 and 12).

3.3 Coreference Resolution for Paragraphs

Different expressions referring to the same entity are often used in text. All pronouns generally refer to some nouns that appeared previously in a given sentence. In this work, we apply CR techniques in order to find anaphors or cataphors in paragraphs, and then substitute them with the original referring entities. This preprocessing can significantly reduce ambiguities in the paragraphs and provide more direct knowledge to BERT. In order to resolve coreferences in the paragraphs, we used the NeuralCoref toolkit [4].[4] NeuralCoref is regarded as a highly extensible model to any new text data. We show a part of a paragraph from SQuAD below:

Paragraph: *Beyoncé Giselle Knowles (born September 4, 1981) is an American singer, songwriter, record producer and actress. Born and raised in Houston,* ***she*** *performed in various singing and dancing competitions as ...*

Resolved Coreference: *Beyoncé Giselle Knowles (born September 4, 1981) is an American singer, songwriter, record producer and actress. Born and raised in Houston,* ***Beyoncé Giselle Knowles*** *performed in various singing and dancing competitions as ...*

In the above example we see that the proper noun ('Beyoncé Giselle Knowles') in the place of the pronoun ('she') reduces ambiguity in the text, which essentially can provide more direct knowledge to the BERT attention model. As above, the input question and modified paragraph are represented as a single packed sequence for the BERT training.

Additionally, we carried out experiments applying multiple preprocessing techniques together to BERT. The intuition is that the contexts from the multiple sources can provide the QA model more reasoning knowledge. The QE (cf. Sections 3.1 & 3.2) and CR (cf. Section 3.3) preprocessing techniques were applied collectively in different combinations.

4 Results and Discussion

4.1 Experimental Setups

This section explains experimental setups including a short overview on the QA data set, SQuAD. SQuAD v1.1 [20] is a standard reading comprehension dataset. It consists

[4] `https://github.com/huggingface/neuralcoref`

of reading paragraphs and associated questions in text format. These paragraphs were taken from Wikipedia articles across the various categories such as history, science etc. An answer to an input question is a segment or span (i.e start and end indices of the segment) from the associated paragraph. The dataset is divided into a training set and a validation set. The training set includes 18,896 paragraphs from 442 documents, which also contains 87,599 questions. The validation set includes 1,867 paragraphs from 49 documents and contains 10,570 questions. In order to evaluate the performance of the QA systems, we used two evaluation metrics as in [20, 5], which are 'exact match' (EM) and F_1. EM measures the percentage of predictions that match exactly with any one of the ground truth answers. F_1 is a measure of the average overlap between the prediction and ground truth answer [20]. We use approximate randomization [27] to test the statistical significance of the difference between two systems. We fine-tuned the BERT models for 3 epochs with a learning rate of *3e-5* as suggested in [5], and set batch size to 32. We followed the recommended best setup by [5] and keep the same setup for all our experiments.

4.2 Evaluation Results

In this section we obtain experimental results to evaluate the performance of our QA systems considering the different preprocessing setups discussed in Section 3. We report the evaluation results in Table 1. As can be seen from the second column of Table 1, our baseline model, $\text{BERT}_{\text{BASE}}$, is quite competitive as it produces an F_1 score of 88.5 and an EM of 80.8 points on the development set.

The third column of Table 1 represents results that we obtained by applying our first preprocessing technique (i.e. QE with WordNet; cf. Section 3.1) to BERT. We call the QA system that incorporates this feature BERT_{WN}. As can be seen from the table, BERT_{WN} outperforms $\text{BERT}_{\text{BASE}}$ in the answer span prediction task (with absolute improvements of 0.3 F_1 and 0.2 EM points over $\text{BERT}_{\text{BASE}}$; however, the improvements are not statistically significant). The fourth column of Table 1 presents evaluation results

Table 1. Evaluation results (EM and F_1 scores) obtained with different QA models.

	$\text{BERT}_{\text{BASE}}$	BERT_{WN}	BERT_{NMT}	BERT_{CR}	BERT_{3F}
F_1	80.8	81.1 ($p > 0.05$)	81.2 ($p > 0.05$)	81.6 ($p < 0.05$)	82.7 ($p < 0.01$)
EM	88.5	88.7 ($p > 0.05$)	88.9 ($p > 0.05$)	89.3 ($p < 0.05$)	89.8 ($p < 0.01$)

that we obtained by integrating the NMT-based QE feature into BERT (cf. Section 3.2). As mentioned in Section 3.2, we carried out experiments integrating varying sizes of alternative questions (n: 3, 5, 7 and 12). As far as the answer span prediction quality by the QA systems is concerned, we found that the setup with the alternative question sequences of size 12 is more effective than the other setups (i.e. with $n = 3, 5, 7$). We call the QA system that includes the NMT based QE feature (with $n = 12$) BERT_{NMT}. We see from Table 1 that BERT_{NMT} outperforms $\text{BERT}_{\text{BASE}}$ in the answer span prediction task (with absolute improvements of 0.4 F_1 and 0.4 EM points over $\text{BERT}_{\text{BASE}}$; however,

the improvements are not statistically significant). The fifth column of Table 1 represents the QA model that incorporates the CR-based features (cf. Section 3.3). We call this QA system BERT_{CR}. BERT_{CR} statistically significantly outperforms $\text{BERT}_{\text{BASE}}$ in the answer span prediction task as per the scores obtained on the development set (the absolute improvements of 0.8 F_1 and 0.8 EM points over $\text{BERT}_{\text{BASE}}$).

Since we found that BERT_{WN}, BERT_{NMT} and BERT_{CR} proved to be effective in the answer span prediction task, we carried out a few more experiments by integrating multiple features collectively into BERT. The model that includes three features collectively (i.e. QE (WordNet) + QE (NMT) + CR features) is found to be the best-performing QA system. This QA system is referred as BERT_{3F}. As can be seen from the last column of Table 1 that BERT_{3F} produces 89.8 F_1 points and 82.7 EM points on the development set (with absolute improvements of 1.3 F_1 and 1.9 EM points over $\text{BERT}_{\text{BASE}}$; both improvements are statistically significant).

4.3 Prediction Analysis

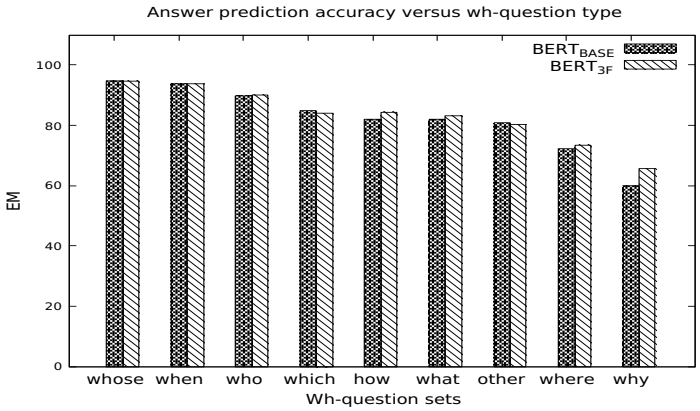
This section presents a comparative analysis of the ability of the BERT QA systems to predict the answers correctly on SQuAD. In order to carry out the analysis, we considered a variety of criteria: (i) wh-question type, (ii) wh-word position in questions, (iii) ground truth answer span size in paragraph, and (iv) data domain, and investigate their relatedness to the QA system's answer prediction quality. This analysis helps us achieve our two goals: (a) unraveling the strengths and weaknesses of BERT on QA, and (b) comparing BERT on two experimental setups: the vanilla baseline (i.e. $\text{BERT}_{\text{BASE}}$) and context-sensitive QA (i.e. BERT_{3F}) models.

Wh-question Type We wanted to see whether there is any relationship between the wh-question type and the performance of the QA models. For this, we considered the commonly used *wh-words* that are used to introduce questions: 'when', 'who', 'which', 'how', 'what', 'where', 'why', and 'whose'. For a given input question, its wh-question type is identified using a simple rule-based procedure. We also have a particular wh-question type ('other') whose questions contain none of the wh-words listed above. We divide the development set examples as per the wh-question type. Thus, we obtained a number of subsets, and each subset contains a particular type of wh-question. From now on, we call such subsets *wh-sets*. For each of the wh-sets we obtain the number of wrong and right answer predictions by $\text{BERT}_{\text{BASE}}$ and BERT_{3F}. In Figure 1, we plot histogram distributions of answer prediction accuracy (EM scores) over the wh-sets.

As can be seen from Figure 1, both QA systems did not perform uniformly across the wh-sets. They performed excellently for predicting answers of 'whose', 'when' and 'who' questions. We also see that both BERT QA models performed moderately on the 'which', 'how', 'other' and 'what' wh-sets, and quite poorly on the 'where' and 'why' wh-sets. When we compare the bars, we see BERT_{3F} outperforms $\text{BERT}_{\text{BASE}}$ in most cases bar two instances (i.e. 'other' and 'which' question types).

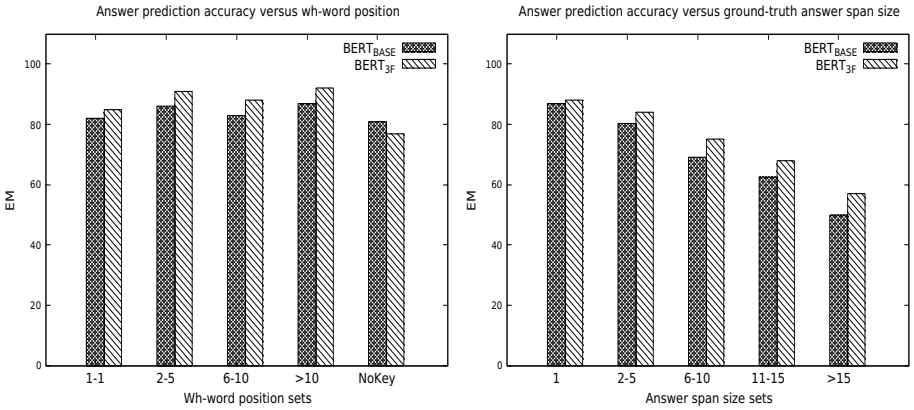
Wh-word Position We wanted to examine whether there is any correlation between the wh-word positions in questions and the performance of BERT in the QA task. For this,

Fig. 1. Correlation between the wh-question types and BERT's answer prediction performance.



we first identify the position of the wh-word in a given input question. As above, we divide the development set examples based on the positions of the wh-words in questions. This creates several subsets, and each subset contains questions whose wh-words appear in a specific position range in the questions (e.g. 1st position, 2nd to 5th position). From now, we call such subsets *wh-pos-sets*. As above, we plot the distributions of the EM scores over the wh-pos-sets in Figure 2a for $\text{BERT}_{\text{BASE}}$ and BERT_{3F}. The x-axis and y-axis of Figure 2a represent the distributions of the EM scores and the wh-pos-sets, respectively. We can see from Figure 2a that no strong relationship can be seen between the wh-word positions in questions and the QA systems' answer prediction quality. As far as the comparison of the performances of $\text{BERT}_{\text{BASE}}$ and BERT_{3F} is concerned, as above, BERT_{3F} outperforms $\text{BERT}_{\text{BASE}}$ on all wh-pos-sets bar one set that contains the questions that have no wh-words.

Ground Truth Answer Span Size This time, we choose a feature from paragraphs for analysis, which is ground truth answer span size. We divide the development set examples based on the number of words into ground truth answers (e.g. one word, two to five words). Thus, we obtained a number of subsets, and each subset contains questions whose answer spans are limited to a range of numbers. From now on, we call such subsets *answer-span-sets*. In Figure 2b, we plot histogram distributions of answer prediction accuracy (EM scores) over the answer-span-sets for $\text{BERT}_{\text{BASE}}$ and BERT_{3F}. The x-axis and y-axis of Figure 2b represent the EM scores and the answer-span-sets, respectively. We can see from Figure 2b that there is a clear relationship between the both QA models' performance and the ground truth answer span size. The answer prediction accuracy declines linearly with the growing number of words in the ground truth answers that the QA models would have to predict. When we compare $\text{BERT}_{\text{BASE}}$ and BERT_{3F} with respect to this feature, we see from Figure 2b that BERT_{3F} outperforms $\text{BERT}_{\text{BASE}}$ in all cases (i.e. on all answer-span-sets).



(a) EM scores versus wh-pos-sets. (b) EM scores versus answer-span-sets.

Fig. 2. Correlation between wh-pos- and answer-span-sets and BERT's performance on QA.

Wikipedia Titles As mentioned in Section 4.1, the development set includes 1,867 paragraphs from 49 documents (the Wikipedia titles). The Wikipedia documents were taken from a variety of domains (e.g. sports, environment, history, engineering, science). We examined our QA models' answer prediction ability on different domains. We found that $\text{BERT}_{\text{BASE}}$ and BERT_{3F}, performed quite well with some specific Wikipedia titles such as 'American_Broadcasting_Company' (EM scores of 95.1 and 96.7, respectively) and 'Steam_engine' (EM scores of 92.1 and 94.5, respectively). We also observed the opposite picture with some of the Wikipedia titles such as 'Packet_switching' (EM scores of 47.2 and 58.5 with $\text{BERT}_{\text{BASE}}$ and BERT_{3F}, respectively). Adapting a model to a specialised domain is seen as a challenging task in many NLP areas. We see that the BERT models (both $\text{BERT}_{\text{BASE}}$ and BERT_{3F}) struggled to deal with the specialised and complex domain data (e.g. computer network) as well as the mixture of multiple domain data (e.g. administration, history and legal). However, we also observed that BERT_{3F} performed better than $\text{BERT}_{\text{BASE}}$ on the specialised and complex domain data most of the time. In addition to the above analysis, we manually looked at a sample of answer prediction examples from the development set. A few of the examples with an analysis on the performance of our context-aware QA systems and $\text{BERT}_{\text{BASE}}$ are made available online.[5]

5 Conclusion

In this work, we presented two automatic QE strategies in QA. As far as our first QE technique is concerned, we first identified good terms from the input questions following a state-of-the-art term classification method, and then used WordNet in order to obtain

[5] `https://github.com/rejwanul-adapt/BERT-analysis/blob/master/Examples-BERT.pdf`

synsets for each of the good terms. We presented a method that applying two word-to-word semantic similarity measures together extracts the most relevant synonyms from the synsets. As far as our second QE method is concerned, we used a state-of-the-art neural MT system in order to produce a set of alternative questions for each input question. Both QE strategies were effective in predicting answers in the QA tasks, although the improvements obtained by the QA systems with the addition of these features over the baseline are not statistically significant. This study also investigated the possibility of applying CR techniques on the paragraphs in QA. The QA model with the CR method significantly outperformed $\text{BERT}_{\text{BASE}}$, with the absolute improvements of 0.8 F_1 and 0.8 EM points over $\text{BERT}_{\text{BASE}}$.

Furthermore, we conducted a number of experiments by integrating multiple features collectively into the BERT model in various combinations. We found that the QA model that integrates all three features (two QE and CR methods) together is the best-performing system as per the F_1 and EM scores. With this setup, the BERT QA system produced significant gains in F_1 and EM (absolute improvements of 1.3 F_1 and 1.9 EM points) over $\text{BERT}_{\text{BASE}}$.

In sum, as far as the QA task on the state-of-the-art BERT architecture is concerned, all our three preprocessing methods are shown to be effective. Most importantly, the gains were achieved (some of them are statistically significant) by applying these methods without making any modification to the model architecture.

Additionally, we carried out a thorough error analysis on the predictions to see how the BERT models (the baseline and our best-performing) performed on QA. In order to do this, we took a variety of criteria into account and examined their relatedness to the answer prediction errors. From our analysis we found that the patterns of the answer prediction errors of the both baseline and our best-performing QA models are nearly similar in most cases. The both BERT QA models performed excellently for certain wh-question types (e.g. 'whose', 'when' and 'who'), although their performances were found to be below par for certain wh-question types (e.g. 'why' and 'where'). As far as the position of wh-words in questions is concerned, we could not find any strong correlation between this feature and answer prediction ability. As for the ground truth answer span size, we found that the answer prediction accuracy declines linearly with the increasing number of words in the ground truth answers that the QA system would have to predict. As far as the above three criteria (wh-question type, wh-word position in questions, answer span size) and systems' answer span prediction accuracy are concerned, our best-performing QA model outperformed the BERT baseline in all cases barring few exceptions. From our analysis we also found that the BERT baseline and our best-preforming QA systems performed below par on certain specialised domain data (e.g. computer network) or the mixture of multiple domain data (e.g. administration, history and legal). However, we observed that the best-performing system performed better than $\text{BERT}_{\text{BASE}}$ on the specialised and complex domain data. This thorough error analysis, to a certain extent, identifies patterns of the examples for which the BERT models tend to make wrong or right predictions in the QA task, which, we believe would help the NLU researchers to fix problems of the model in relation to this task.

As mentioned in Section 4.3, our WordNet-based QE method expands a good term by generating its relevant synonyms, which, however, may not be the same morphologi-

cal forms as the good term is as the QE method does not have morphological generation module. In future, we intend to add a morphological generation module in this QE technique. We also intend to carry out a deeper analysis on BERT considering more criteria, e.g. length of the questions, head versus tail questions, and comparing the BERT models with the classical IR models.

Acknowledgments

The ADAPT Centre for Digital Content Technology is funded under the Science Foundation Ireland (SFI) Research Centres Programme (Grant No. 13/RC/2106) and is co-funded under the European Regional Development Fund. This project has partially received funding from the European Union's Horizon 2020 research and innovation programme under the Marie Skłodowska-Curie grant agreement No. 713567, and the publication has emanated from research supported in part by a research grant from SFI under Grant Number 13/RC/2077.

References

1. Peter Cahill, Jinhua Du, Andy Way, and Julie Carson-Berndsen. Using same-language machine translation to create alternative target sequences for text-to-speech synthesis. In *Proceedings of Interspeech 2009, the 10th Annual Conference of the International Speech Communication Association*, pages 1307–1310, Brighton, UK, 2009.
2. Guihong Cao, Jian-Yun Nie, Jianfeng Gao, and Stephen Robertson. Selecting good expansion terms for pseudo-relevance feedback. In *Proceedings of the 31st Annual International ACM SIGIR Conference on Research and Development in Information Retrieval*, pages 243–250, Singapore, 2008.
3. Claudio Carpineto and Giovanni Romano. A survey of automatic query expansion in information retrieval. *ACM Computing Surveys (CSUR)*, 44(1):1, 2012.
4. J. Chaumond. Fast coreference resolution in spaCy with neural networks; available online: `https://github.com/huggingface/neuralcoref`; accessed on 8-october-2019.
5. Jacob Devlin, Ming-Wei Chang, Kenton Lee, and Kristina Toutanova. BERT: pre-training of deep bidirectional transformers for language understanding. *CoRR*, abs/1810.04805, 2018.
6. Renxu Sun Jing Jiang Yee Fan, Tan Hang Cui Tat-Seng Chua, and Min-Yen Kan. Using syntactic and semantic relation analysis in question answering. In *Proceedings of the 14th Text REtrieval Conference (TREC)*, 2005.
7. Federico Fancellu, Morgan O'Brien, and Andy Way. Standard language variety conversion using smt. In *Proceedings of the Seventeenth Annual Conference of the European Association for Machine Translation*, pages 143–149, Dubrovnik, Croatia, 2014.
8. Sepp Hochreiter and Jürgen Schmidhuber. Long short-term memory. *Neural Computation*, 9:1735–1780, 12 1997.
9. Mohit Iyyer, Jordan L. Boyd-Graber, Leonardo Max Batista Claudino, Richard Socher, and Hal Daumé. A neural network for factoid question answering over paragraphs. In *EMNLP 2014: Proceedings of the Conference on Empirical Methods in Natural Language Processing*, pages 633–644, Doha, Qatar, 2014.
10. Marcin Junczys-Dowmunt, Roman Grundkiewicz, Tomasz Dwojak, Hieu Hoang, Kenneth Heafield, Tom Neckermann, Frank Seide, Ulrich Germann, Alham Fikri Aji, Nikolay Bogoychev, André F. T. Martins, and Alexandra Birch. Marian: Fast neural machine translation in C++. In *Proceedings of the ACL, System Demonstrations*, pages 116–121, Melbourne, Australia, 2018.

11. Philipp Koehn. Europarl: A parallel corpus for statistical machine translation. In *Proceedings of the Machine Translation Summit X*, pages 79–86, Phuket, Thailand, 2005.
12. Ruslan Mitkov. *Anaphora Resolution*. Longman, Harlow, UK, 2002.
13. Thomas S. Morton. Using coreference for question answering. In *Proceedings of the ACL Workshop on Coreference and Its Applications*, pages 85–89, College Park, MD, 1999.
14. Peter Oram. WordNet: An electronic lexical database. *Applied Psycholinguistics*, 22:131–134, 2001.
15. Kishore Papineni, Salim Roukos, Todd Ward, and Wei-Jing Zhu. BLEU: A method for automatic evaluation of machine translation. In *Proceedings of the 40th Annual Meeting on Association for Computational Linguistics*, pages 311–318, Philadelphia, PA, 2002.
16. Ellie Pavlick, Pushpendre Rastogi, Juri Ganitkevitch, Benjamin Van Durme, and Chris Callison-Burch. PPDB 2.0: Better paraphrase ranking, fine-grained entailment relations, word embeddings, and style classification. In *Proceedings of the 53rd Annual Meeting of the Association for Computational Linguistics and the 7th International Joint Conference on Natural Language Processing (Volume 2: Short Papers)*, pages 425–430, Beijing, China, 2015.
17. Matthew Peters, Mark Neumann, Mohit Iyyer, Matt Gardner, Christopher Clark, Kenton Lee, and Luke Zettlemoyer. Deep contextualized word representations. In *Proceedings of the Conference of the North American Chapter of the Association for Computational Linguistics: Human Language Technologies*, pages 2227–2237, New Orleans, LA, 2018.
18. Vassilis Plachouras, Fabio Petroni, Timothy Nugent, and Jochen L. Leidner. A comparison of two paraphrase models for taxonomy augmentation. In *Proceedings of the 2018 Conference of the North American Chapter of the Association for Computational Linguistics: Human Language Technologies, Volume 2 (Short Papers)*, pages 315–320, New Orleans, LA, 2018.
19. Yifan Qiao, Chenyan Xiong, Zheng-Hao Liu, and Zhiyuan Liu. Understanding the behaviors of BERT in ranking. *CoRR*, abs/1904.07531, 2019.
20. Pranav Rajpurkar, Jian Zhang, Konstantin Lopyrev, and Percy Liang. SQuAD: 100,000+ questions for machine comprehension of text. In *EMNLP 2016: Proceedings of the Conference on Empirical Methods in Natural Language Processing*, pages 2383–2392, Austin, TX, 2016.
21. Marc-Antoine Rondeau and Timothy J Hazen. Systematic error analysis of the stanford question answering dataset. In *Proceedings of the Workshop on Machine Reading for Question Answering*, pages 12–20, 2018.
22. Roland Stuckardt. Coreference-based summarization and question answering: a case for high precision anaphor resolution. In *Proceedings of the 2003 International Symposium on Reference Resolution and Its Applications to Question Answering and Summarization*, pages 33–42, Venice, Italy, 2003.
23. Betty van Aken, Benjamin Winter, Alexander Löser, and Felix A Gers. How does bert answer questions? a layer-wise analysis of transformer representations. In *Proceedings of the 28th ACM International Conference on Information and Knowledge Management*, pages 1823–1832, 2019.
24. Ashish Vaswani, Noam Shazeer, Niki Parmar, Jakob Uszkoreit, Llion Jones, Aidan N. Gomez, Lukasz Kaiser, and Illia Polosukhin. Attention is all you need. *CoRR*, abs/1706.03762, 2017.
25. José L. Vicedo and Antonio Ferrández. Importance of pronominal anaphora resolution in question answering systems. In *Proceedings of the 38th Annual Meeting of the Association for Computational Linguistics*, pages 555–562, Hong Kong, 2000.
26. Z. Wu and M. Palmer. Verb semantics and lexical selection. In *Proceedings of the 32nd Annual meeting of the Associations for Computational Linguistics*, pages 133–138, Las Cruces, NM, 1994.

27. Alexander Yeh. More accurate tests for the statistical significance of result differences. In *Proceedings of the 18th Conference on Computational Linguistics*, pages 947–953, Saarbrücken, Germany, 2000.