Simple Question Answering Over a Domain-Specific Knowledge Graph using BERT by Transfer Learning

Mani Vegupatti[1], Matthias Nickles[1], and Bharathi Raja Chakravarthi[2]

[1] School of Computer Science, National University of Ireland, Galway
[2] Insight SFI Research Centre for Data Analytics, Data Science Institute, National University of Ireland, Galway

Abstract. We build and evaluate a baseline for simple question answering over a domain-specific knowledge graph by using a pretrained open-domain language model BERT. Training a neural network from scratch needs a large annotated dataset whereas transfer learning adapts a pretrained language model and allows task-specific fine-tuning with limited-data. However, building a domain-specific language model needs a large amount of domain-specific text, resource, and time for pretraining. But open-domain language models such as BERT are readily available for use. Hence, we evaluate the open-domain pretrained BERT for creating a domain-specific question answering baseline model that requires less amount of training data. In this work, we built a BioMed domain simple question answering system by fine-tuning the open-domain BERT with a manually curated dataset of ˜600 questions from the Drugbank knowledge graph published by Bio2RDF.

Keywords: Knowledge Graph · Question Answering · BERT · Transfer Learning

1 Introduction

Question Answering (QA) is one of the earliest research interests in artificial intelligence started from answering questions posed in natural language based on underlying database data[7, 26], extracting answer from the given text passage[24] and recently focused on QA over a Knowledge Graph (KG). KG represents the facts about entities as a graph, where the nodes represent the entities which can be real-word persons, places, objects, concepts, events, and many other and edges link the entities and serve as a predicate[9]. QA over a KG aims at providing answer for a natural language question by using the facts from the KG and the two categories are Simple/Factoid QA and Complex QA[2, 22]. Simple QA is called simple not because the QA task is simple, but the answering process requires simple reasoning processing involving only a single triple from a KG[2]. Since the question can be answered using a single fact of a KG, it is also known as single-factoid QA[17]. Complex QA requires a complex reasoning process with hops over multiple triples of a KG to retrieve the answer[22].

Training a neural network from scratch for QA needs a large volume of annotated data and the creation of such dataset requires a lot of time and resources that are scarce. However, transfer learning pretrains a Language Model (LM) to learn task-independent context-based language representations from large unannotated text and allows finetuning for a downstream task with limited training data. Transfer learning-based approaches use the pretrained language models such as ELMO, ULMFiT, GPT and BERT to achieve better performance in GLUE tasks with less amount of data and fewer epochs[15, 10, 18, 3].

KGs can be classified into open-domain and domain-specific. Open-domain KG is a very large collection of coarse-grained facts without restriction to any specific domain whereas domain-specific KG is relatively smaller size with fine-grained facts dedicated to a single domain like life sciences, academic and tourism [4]. Increasing adoption of KGs in industry and multiple domains[9] fuels the necessity of QA over a domain-specific KG. Our focus in this work is to adapt an open-domain pretrained language model like BERT for domain-specific simple QA over a KG in transfer learning settings and build a baseline model architecture.

We choose the biomedical domain of life sciences because of its complexity and importance. The terminologies of biomedical are complex and significantly different than open-domain. This will help to evaluate the effectiveness of open-domain BERT adoption to the domain-specific QA task. Biomedical QA is essential in improving health care and its growing importance attracted multiple QA challenges, but there exists no dataset for simple QA over a biomedical KG (refer Section 3). Hence, we created Drugbank simple question answering dataset using the facts from Drugbank KG released by Bio2RDF[5]. Main contributions of our work are:

- Creation of Drugbank Simple Question Answer Dataset (Drugbank SQA) for the task simple QA over a domain-specific KG and annotations for subtasks Named Entity Recognition (NER) and classification.
- Building and evaluating the baseline model architecture for answering Simple Question from the facts of a domain-specific KG using pretrained open-domain language model BERT in transfer learning settings.
- Presenting the evaluation of various techniques for adaptation of open-domain pretrained BERT LM in simple QA over a domain-specific KG.

2 Related Work

The methods used for simple QA over a KG can be broadly classified into end-to-end neural networks, baseline models and transformer-based models. End-to-end neural network models employ a RNN-based single complex deep neural network for the whole task[1, 20, 2] and often transform inputs using word or character level embeddings [8, 13]. Baseline models divide the QA task into subtasks and use simple models for conquering individual subtasks[23, 17, 14]. However, both the approaches need a large amount of labeled data and depend on the sequence modeling that increases the training time.

Transformer-based models use only attention mechanism and remove RNN from the architecture. These models achieve global long-term dependencies and generalization by learning task-independent features[12]. Transformer based GPT and BERT are shown to out-perform end-to-end neural networks in simple QA over text passages using a pretrained language model and fine-tuning with task-specific data[18, 3]. BERT performs better than the former since it uses bi-directional information along with attention[3] for learning representations.

BERT[3] is designed to pre-train deep bidirectional representations from an unlabeled text by jointly conditioning on both right and left context in all layers. The pre-trained BERT model is finetuned with just a single additional output layer to create models for a wide range of the task, such as question answering. Various transfer learning approaches for using BERT in the downstream tasks like NER and classification and their effectiveness are studied in[3, 16]. Simple QA over an open-domain KG using BERT was carried out with results outperforming earlier Bi-LSTM models [12].

3 DrugBank Simple Question Answering Dataset

In biomedical question answering, TREC Genomics Track[3] and QA4MRE[4] are the datasets for QA without KGs over a passage of text, QALD[5] aims at QA over interlinked datasets (SIDER, diseasome and drugbank) and BioASQ[6] targets answering questions by combing various heterogeneous sources like texts, databases, and triple stores[25]. But we do not have a dataset for simple QA over a KG, hence we created a new dataset[7] of 566 questions out of ˜3 million facts available in the DrugBank KG.

We created the dataset with a three-step process. First, we examined the pattern of 3670K triples in the KG then eliminated the triples created for structuring the KG like types and properties. Finally, to ensure enough coverage distinct relations were selected from available relations and questions were generated with English variation of the selected relations. Table 1 shows a few sample questions.

Table 1. Sample Question

Question	Triple (answer in bold font)
Provide the estimated half life for Fusidic Acid?	(Fusidic Acid, half life, **Approximately 5 to 6 hours in adults**)
Who produces Penicillin V?	(Penicillin V, manufacturer, **Eli lilly and co**)
which organisms are impacted by Fluspirilene?	(Fluspirilene, affected-organism, **Humans and other mammals**)

[3] https://trec.nist.gov/data/genomics.html
[4] http://nlp.uned.es/clef-qa/repository/qa4mre.php
[5] http://qald.aksw.org/4/documents/qald-4.pdf
[6] http://bioasq.org/
[7] https://github.com/mani-vegupatti/SQA_Over_DrugBank_KG/tree/master/dataset

4 Methodology

In our work, we will use a domain-specific knowledge graph, conduct experiments using the architectures inspired from the baseline models[14, 23] for single factoid question-answering in transfer learning settings with open-domain pretrained BERT language model.

4.1 Architecture

We use the architecture of baseline model approach that divides the simple QA task into subtasks subject identification, relation classification and subject linking. This approach helps in data reduction, understandability and choosing the best architecture for individual subtasks. Different models are built for individual subtasks hence we can choose task-specific architecture. Simple architecture helps in reducing deep layers and in turn the training data. Finally, this makes it easy to understand the performance of models in individual subtasks and parts of the system. The architecture of the system is shown in below Figure 1.

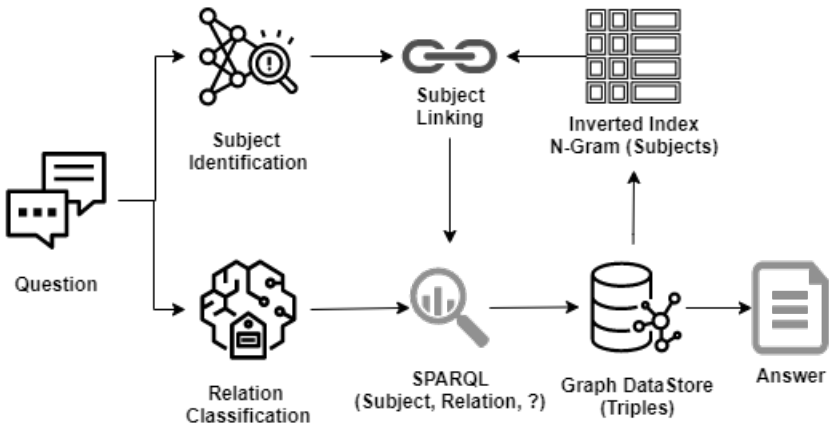

Fig. 1. Architecture - Simple QA Over a Domain-specific KG

- Question: The question is asked in natural language to know a fact regarding an entity in the KG.
  Example: q = 'who produces penicillin V'
- Subject Identification: Subject Identification is the process of predicting the substring from the question phrase that matches the subject/entity of the question.This is the problem of named entity recognition for the given text and can be solved using the sequence labelling or token classification task
  Example: s = 'penicillin V' | q = 'who produces penicillin V'

- Relation Classification: Relation classification is used to predict the correct relation for a given question over the available relations. This is formulated and solved as a classification problem
  Example: r = 'manufacturer' | q = 'who produces penicillin V'
- Graph Datastore: KG is stored as a collection of RDF triples in the graph datastore. We use the n-triple format DrugBank KG released by Bio2RDF.
- Inverted Index: Inverted index is created using n-grams of the entity labels as keys and entity/entity-URI as value.
- Subject Ranking and Linking: The string identified as the subject in the subject identification module is looked up in the inverted index and Top K subjects are selected based on fuzzy search and scoring. Fuzzy search helps to identify the terms based on partial-string matching and scoring by using techniques like Levenshtein distance/edit distance
- Answer Generation: Answer generation is carried out by sending a SPARQL query to the SPARQL endpoint of the given KG. SPARQL query is formed using the subject/entity received from the subject ranking and linking module, and relation/predicate obtained from the relation classification module.

4.2 Adapting BERT in Transfer Learning

The subtasks of simple QA over a KG are solved by formulating them as NER and classification problems. BERT can be adopted for downstream tasks in the transfer learning settings using one of the two approaches feature extraction or fine-tuning. We try both the approaches to find the best approach for the sub-tasks (refer section 4.3 and 4.4).

- Feature Extraction: In the feature extraction approach, we will extract the pretrained representations from the BERT model and use them as features for the downstream tasks. Alternatively, pretrained layers with its weights can be used as-is without fine-tuning. The advantages[3] of this approach are:
  - For the tasks that can not be solved by the transformer architecture, performance can be improved by using task-specific architectures with contextualised BERT embeddings
  - It is computationally less expensive when the input pretrained representations are not further changed during fine-tuning.
- Fine-tuning: In the fine-tuning approach, a task-specific classification layer is added on top of the pretrained model. The parameters of the classification layer are learned along with adjustment of the pretrained parameters of the underlying layers while training on the required objective of the given downstream task

4.3 Subject Identification

We built the module subject identification based on the concept named entity recognition, which predicts the span of a given text that identifies the entity and

its type[11]. This module uses Begin-Inside-Outside (BIO) system is for tagging the tokens of the given sentence during the learning/prediction. We want to identify the chunk of words representing the subject/entity and do not want to classify the type of subject, hence the final tags used are B-E (Beginning of an entity), I-E (Inside the entity) and O-E (Outside the entity).

We have built three models, two based on feature extraction (S1, S3) and one using the fine-tuning approach (S2) and selected the best model for building the final pipeline.

S1 - Sequence Labelling with BiLSTM + CRF using BERT word embeddings:
Previous works[17, 14] on building baseline models for simple QA over an open-domain KG achieved the top score by using BiLSTM+CRF in the NER task. Hence for comparison purpose, we built the BiLSTM+CRF sequence labelling model using BERT word embeddings. BERT provides context-based word embeddings based on the local context in which the word appears and help to overcome problems like polysemy. This model employes a feature extraction approach in which the word embeddings from the pretrained BERT LM is used as input to LSTM encoder and final tagging is obtained from CRF decoder.

S2 - Token Classifier with fully fine-tuned BERT layers:
We formulated the problem as a token classification and built the model using the fine-tuning approach. In this design, the pretrained BERT model is used as a base for providing input to the classifier layer which is a linear classifier with softmax activation. While fine-tuning the model on the downstream NER task, all parameters of the pretrained BERT model are also modified along with the parameters of the linear classifier with the token classification objective function. The cross-entropy loss function is used for training and the probability of the token calculated as below,

$$P(t|h_i) = softmax(w_i h_i + b) \tag{1}$$

$$\begin{aligned} where, \\ t \in \{B\text{-}E, I\text{-}E, O\} \\ h_i = input\ to\ classifier\ from\ BERT \\ w_i,\ b = learned\ weight\ and\ bias \end{aligned}$$

S3 - Token Classifier with frozen BERT layers:
This model design and architecture are the same as the previous model with an exception that the base layers of BERT are fully frozen, which means the pretrained weights of the base layers are not adjusted during the fine-tuning process.

4.4 Relation Classification

Relation classification is the module for predicting the right relation/predicate for the given question from the list of relations obtained from the KG that

connects the subject with the object. In a simple QA task, it is assumed that each question will have at max only one relation for any given question. It is solved using multiclass sequence/text classification i.e. for the given sequence of words, predict the best class(relation) that represents the given question from the available classes. It is formulated as below,

$$P(r_i \mid [x_1\ x_2\ x_3\ ...\ x_n]) \tag{2}$$

$$\begin{aligned} &where, \\ r_i &= relation_i \in R \\ R &= set\ of\ relations: \ \{r_1\ r_2\ r_3\ ...\ r_n\} \\ [x_1\ x_2\ x_3\ ...\ x_n] &= Question: \ A\ sequence\ of\ words \end{aligned}$$

We selected the best model for the final pipeline from the four models we built of which three (R1, R2 and R4) are based on feature extraction and one (R3) is based on fine-tuning approach.

R1 - SVM Classifier with BERT sentence embeddings:
In this model, we want to find the effectiveness features extracted from BERT language model when used as input to classical machine learning algorithms. We conducted experiments with various conventional ML algorithms and found SVM produces the best results. Sentence embeddings from BERT LM provides the complete semantic representation of the sentence which can be used for the classification task. Hence, we build the model by extracting sentence embeddings as the feature vector from BERT output and passing it as input to the conventional ML classifier. With this approach, we avoid the manual feature extraction for the given text instead use the embeddings (representation of the question) from the BERT LM.

R2 - BiLSTM Classifier with BERT word embeddings:
In earlier works[19, 14], BiLSTM architecture was used for building the classification task of simple QA over an open-domain KG that produced top score. For comparison purpose, we also build feature extraction based BiLSTM classifier. Since LSTM requires sequence-based inputs instead of sentence embeddings we used word embeddings as input to this model. We added a classifier of a dense layer with softmax activation on top of the LSTM layer which produces the output i.e., the probability of the relations.

R3 - Relation Classifier with fully fine-tuned BERT layers:
The problem is formulated as a multiclass classification of a text sequence (sentence) and the representation of the sentence is obtained from the last hidden layer of the pretrained BERT model. The [CLS] token of BERT output captures the complete syntactic and semantic representation of the sentence based on the language model trained on the masked token

and next sentence prediction. The output at [CLS] token is passed as an input to the next fully connected dense layer with the softmax activation which serves as the classification layer. the weights of the fully connected layer and the base layers are jointly adjusted while finetuning the model on the multiclass classification task with task-specific data and categorical cross-entropy loss function

R4 - Relation Classifier with fully frozen BERT layers:
In this model, the base layers adapted from the pretrained BERT LM is fully frozen. While fine-tuning, the pretrained layer's weights are not adjusted and only weights of top classifier layer are updated

4.5 Subject Ranking and Linking

The substring of words returned from the question by the NER task can be an exact or partial match of the actual entity. We use this module to find the actual entity. In this module, we created an inverted index that has n-grams of entities as dictionary terms with entities as a posting list and used the Fuzzy-Wuzzy package[8] to search the actual entity from the inverted index based on the substring match of predicted entity string in the dictionary terms.

4.6 Answer Generation

We find answer for given a question from the KG by formulating the SPARQL query using the subject(s) and relations(r) returned by the previous modules subject liking and relation classification respectively. The answer(object) is retrieved using the query "SELECT ?object where s p ?object".

5 Experimental settings and Evaluations

5.1 Model Settings

We have used the pretrained BERT language model 'bert-base-uncased'[9] from Huggingface Transformers to build all the seven models and details are as below:

- Input data split and preprocessing: We have retained 20 per cent of data from Drugbank SQA dataset as test data. Remaining 80 per cent data is further split into training and validation with 80:20 ratio. We used stratification to retain the class balance across datasets. The numbers of examples in training, validation, and testing datasets are 406, 46, and 116 respectively. We tokenised the questions with word piece model[3], added special tokens [CLS] at the start and [SEP] at the end.

[8] https://github.com/seatgeek/fuzzywuzzy
[9] https://huggingface.co/transformers/v2.4.0/model doc/bert.html

- Feature extraction approach: The feature extraction based models used word embeddings, sentence embeddings or frozen pretrained layers. The word vector is constructed by summing the vectors of word pieces and word embedding is obtained by adding the last four layers. The sentence embedding is obtained from [CLS] token position of last hidden state. When pretrained layers are used for feeding the input feature, their weights are not updated during the fine-tuning process.
- Fine tunning approach: The pretrained layers of BERT were also fine-tuned along with the top task-specific layer using the Drugbank SQA training data with task-specific objective function.

5.2 Results

For the subtasks subject identification and relation classification, we have built models with BERT using both transfer learning approaches feature extraction and fine-tuning. We used F-Score for entity-level evaluation of NER tasks[21] and accuracy for relation classification and final entity-relation pair predictions[6]. In both the tasks, fully fine-tuned models outperform the feature extraction based models as shown in Table 2 and 3.

Table 2. Evaluation Results: NER Models

Model	Accuracy	F-Score
Feature extraction based models		
Bi-LSTM+CRF (BERT Word Embeddings)	93.5%	37.1%
Frozen BERT Layers	90.4%	81.0%
Fine-tuning based models		
Finetuned BERT Layers	98.1%	95.5%
Finetuned BERT Layers with FuzzySearch	NA	99.1%

Table 3. Evaluation Results: Relation Classification Models

Model	Result
Feature extraction based models	
SVM (BERT Sentence Embeddings)	64.9%
Bi-LSTM (BERT Word Embeddings)	68.4%
Frozen BERT layers	43.0%
Fine-tuning based model	
Finetuned BERT layers	96.5%

The module entity linking which uses fuzzy-search improves the entity-level accuracy of the NER and in turn increases the final accuracy of the (entity, relation) pair. The final answer (entity-relation pair) prediction accuracy of our model for simple QA over a domain-specific KG along with results of various approaches of open-domain QA are shown in below in Table 4.

Table 4. Final Results

Approach	Accuracy
Simple QA over an open-domain KG - Dataset [2]	
Bi-LSTM + Bi-GRU [14]	74.9%
Bi-LSTM + CNN [14]	74.7%
Bi-LSTM-CRF + BiLSTM [17]	78.1%
BERT [12]	77.3 %
Simple QA over a domain-specific KG - Our dataset DrugBank SQA	
Fully fine-tuned BERT	92.9%
Fully fine-tuned BERT with FuzzySearch	95.6%

5.3 Discussion

In NER task, earlier work on open-domain QA with a large training dataset (75.9 K training examples) reported 91% and 89.8% F-Score using BiLSTM and CRF models respectively[14]. In domain-specific task with a training dataset of ˜400 examples, this architecture with BERT word embeddings could achieve only 37.1% because the training data is not sufficient to learn the ˜50000 parameters and open-domain word embeddings have difficulty in recognising the domain-specific entities. Another feature extraction based model that uses frozen pretrained layers can reach 81.0% but still lower than score 95.5% of the fully fine-tuned model because open-domain trained frozen layers still do not fully recognise domain-specific entities.

In relation classification, feature extraction based BiLSTM with BERT word embeddings model (68.4%) outperforms rest of feature extraction based models SVM with BERT sentence embeddings (64.9%) and frozen BERT layers (43.0%). However, the top score 96.5% is achieved by fully fine-tuned BERT model. This again indicates the inability of the open-domain BERT models to understand domain-specific terms without further fine-tuning with domain-task-specific data.

The open-domain QA reference models use a large dataset of ˜100K questions from freebase[2] whereas our models use the DrugBank SQA dataset of ˜600 questions. Since the datasets used were different, the results are not directly comparable but used for understanding current performance levels. Our research aim is to build a baseline model for domain-specific simple QA by transfer learning from open-domain trained BERT LM for the environments with a scarcity of data, time and resource. With this research aim, our baseline architecture is able to achieve State Of The Art (SOTA) results (95.6% accuracy for entity-relation pair prediction) with fully-fine tuned BERT models for both the subtasks and entity-linking with fuzzy-search.

6 Conclusion

To the best of our knowledge, this work is the first baseline model pipeline for answering a simple question over a domain-specific KG by using open-domain

trained LM BERT in transfer learning settings. We have contributed further by creating the Drugbank SQA dataset by using facts from the DrugBank KG and annotated with required BIO tagging and target classes for the subtasks NER and classification. We have presented an architecture for the baseline domain-specific simple QA model pipeline that contains subtasks subject identification, relation classification, subject-relation linking and answer generation which produces a SOTA result of 95% accuracy for DrugBank SQA dataset. We have also presented multiple methods for adaption of open-domain BERT in domain-specific tasks of QA and evaluated their effectiveness.

References

1. Bordes, A., Chopra, S., Weston, J.: Question answering with subgraph embeddings. In: Proceedings of the 2014 Conference on Empirical Methods in Natural Language Processing (EMNLP). pp. 615–620. ACL, Doha, Qatar (Oct 2014)
2. Bordes, A., Usunier, N., Chopra, S., Weston, J.: Large-scale simple question answering with memory networks. arXiv preprint arXiv:1506.02075 (2015)
3. Devlin, J., Chang, M.W., Lee, K., Toutanova, K.: BERT: Pre-training of deep bidirectional transformers for language understanding. In: Proceedings of the 2019 Conference of the North American Chapter of the ACL, Volume 1 (Long and Short Papers). pp. 4171–4186. ACL, Minneapolis, Minnesota (Jun 2019)
4. Dimitrakis, E., Sgontzos, K., Tzitzikas, Y.: A survey on question answering systems over linked data and documents. Journal of Intelligent Information Systems pp. 1–27 (2019)
5. Dumontier, M., Callahan, A., Cruz-Toledo, J., Ansell, P., Emonet, V., Belleau, F., Droit, A.: Bio2rdf release 3: A larger connected network of linked data for the life sciences. In: Proceedings of the 2014 International Conference on Posters & Demonstrations Track - Volume 1272. p. 401–404. ISWC-PD'14, CEUR-WS.org, Aachen, DEU (2014)
6. Esuli, A., Sebastiani, F.: Evaluating information extraction. In: International Conference of the Cross-Language Evaluation Forum for European Languages. pp. 100–111. Springer (2010)
7. Green Jr, B.F., Wolf, A.K., Chomsky, C., Laughery, K.: Baseball: an automatic question-answerer. In: Papers presented at the May 9-11, 1961, western joint IRE-AIEE-ACM computer conference. pp. 219–224 (1961)
8. He, X., Golub, D.: Character-level question answering with attention. In: Proceedings of the 2016 Conference on Empirical Methods in Natural Language Processing. pp. 1598–1607. ACL, Austin, Texas (Nov 2016)
9. Hogan, A., Blomqvist, E., Cochez, M., d'Amato, C., de Melo, G., Gutierrez, C., Gayo, J.E.L., Kirrane, S., Neumaier, S., Polleres, A., et al.: Knowledge graphs. arXiv preprint arXiv:2003.02320 (2020)
10. Howard, J., Ruder, S.: Universal language model fine-tuning for text classification. In: Proceedings of the 56th Annual Meeting of the ACL (Volume 1: Long Papers). pp. 328–339. ACL, Melbourne, Australia (Jul 2018)
11. Li, J., Sun, A., Han, J., Li, C.: A survey on deep learning for named entity recognition. IEEE Transactions on Knowledge and Data Engineering (2020)
12. Lukovnikov, D., Fischer, A., Lehmann, J.: Pretrained transformers for simple question answering over knowledge graphs. In: International Semantic Web Conference. pp. 470–486. Springer (2019)

13. Lukovnikov, D., Fischer, A., Lehmann, J., Auer, S.: Neural network-based question answering over knowledge graphs on word and character level. In: Proceedings of the 26th international conference on World Wide Web. pp. 1211–1220 (2017)
14. Mohammed, S., Shi, P., Lin, J.: Strong baselines for simple question answering over knowledge graphs with and without neural networks. In: Proceedings of the 2018 Conference of the North American Chapter of the ACL: Human Language Technologies, Volume 2 (Short Papers). pp. 291–296. ACL, New Orleans, Louisiana (Jun 2018)
15. Peters, M., Neumann, M., Iyyer, M., Gardner, M., Clark, C., Lee, K., Zettlemoyer, L.: Deep contextualized word representations. In: Proceedings of the 2018 Conference of the North American Chapter of the ACL: Human Language Technologies, Volume 1 (Long Papers). pp. 2227–2237. ACL, New Orleans, Louisiana (Jun 2018)
16. Peters, M.E., Ruder, S., Smith, N.A.: To tune or not to tune? adapting pretrained representations to diverse tasks. In: Proceedings of the 4th Workshop on Representation Learning for NLP (RepL4NLP-2019). pp. 7–14. ACL, Florence, Italy (Aug 2019)
17. Petrochuk, M., Zettlemoyer, L.: SimpleQuestions nearly solved: A new upperbound and baseline approach. In: Proceedings of the 2018 Conference on Empirical Methods in Natural Language Processing. pp. 554–558. ACL, Brussels, Belgium (Oct-Nov 2018)
18. Radford, A., Narasimhan, K., Salimans, T., Sutskever, I.: Improving language understanding by generative pre-training. URL https://s3-us-west-2. amazonaws. com/openai-assets/researchcovers/languageunsupervised/language understanding paper. pdf (2018)
19. Simperl, E., Norton, B., Acosta, M., Maleshkova, M., Domingue, J., Mikroyannidis, A., Mulholland, P., Power, R.: Using linked data effectively (2013)
20. Sukhbaatar, S., Szlam, A., Weston, J., Fergus, R.: End-to-end memory networks. In: Proceedings of the 28th International Conference on Neural Information Processing Systems - Volume 2. p. 2440–2448. NIPS'15, MIT Press, Cambridge, MA, USA (2015)
21. Tjong Kim Sang, E.F., De Meulder, F.: Introduction to the CoNLL-2003 shared task: Language-independent named entity recognition. In: Proceedings of the Seventh Conference on Natural Language Learning at HLT-NAACL 2003. pp. 142–147 (2003)
22. Trivedi, P., Maheshwari, G., Dubey, M., Lehmann, J.: Lc-quad: A corpus for complex question answering over knowledge graphs. In: International Semantic Web Conference. pp. 210–218. Springer (2017)
23. Ture, F., Jojic, O.: No need to pay attention: Simple recurrent neural networks work! In: Proceedings of the 2017 Conference on Empirical Methods in Natural Language Processing. pp. 2866–2872. ACL, Copenhagen, Denmark (Sep 2017)
24. Voorhees, E.M., Tice, D.M.: Building a question answering test collection. In: Proceedings of the 23rd annual international ACM SIGIR conference on Research and development in information retrieval. pp. 200–207 (2000)
25. Wasim, M., Mahmood, W., Khan, U.G.: A survey of datasets for biomedical question answering systems. International Journal of Advanced Computer Science and Applications **8**(7), 484–488 (2017)
26. Woods, W.A., WA, W.: Lunar rocks in natural english: Explorations in natural language question answering. (1977)