Extractive Question Answering Using Transformer-Based LM

Raj Jha(✉) and V. Susheela Devi

Indian Institute of Science, Bengaluru 560012, India
rajjha@iisc.ac.in, susheela@iisc.ac.in
https://www.iisc.ac.in/

Abstract. Many institutions, organizations, and government bodies deal with a large number of financial documents (which can be structured or unstructured). To avoid the labor-intensive, manual tasks, we propose a Question Answering System in the finance domain to create profitable and competitive advantages for various organizations by making it easier for financial advisors to make decisions. Various pre-trained language models have proven highly effective at extractive question answering. Yet, generalizability stays a challenge for most of these pre-trained language models. In our work, we trained and fine-tuned RoBERTa model on other questions answering datasets of varying difficulty levels to decide which models are competent for generalizing the most thoroughly across varying datasets. Further, we proposed a new methodology to handle long-form answers by modifying the BERT and RoBERTa architecture. We have added the dynamic masking (instead of using static masking) and performed stride-shift (similar to kernel shift in computer vision) in BERT and RoBERTa architecture and compared it with different pre-trained LM to decide if adding dynamic masking and shifting the strides can improve model performance. We have used MRR (Mean Reciprocal Rank), NDCG (Normalized Discounted Cumulative Gain), and Precision@1 to check the performance of our model on FiQA datasets. Moreover, we have used F1-score and Exact Match as performance metrics to set the benchmark for review-based SubjQA datasets. We found out that combining RoBERTa with dynamic masking and stride shift and using Dense Passage Retriever for extracting relevant passages performs the best on both the datasets SubjQA and Financial Question Answer (FiQA) [1,2], and it outperforms the baseline BERT model. The results show an improvement in each metric as measured against the various other models.

Keywords: RoBERTa · Stride Shift · Dynamic Masking

1 Introduction

In the past years, like many other industries/organizations, the financial sector has seen NLP as an ally in better-assisting clients by drawing insightful information such as predicting the company's stocks performance, analyzing 10K

M. Tanveer et al. (Eds.): ICONIP 2022, CCIS 1792, pp. 373–384, 2023.
https://doi.org/10.1007/978-981-99-1642-9_32

reports, assets management related queries, etc. But to answer these queries, financial advisers need to read and analyze a lot of documents. Moreover, the analysis results vary from person to person depending upon their experience level, which sometimes results in inconsistent interpretations of those documents [1]. Therefore, implementing the Question Answering System in financial matters is essential due to the industry's highly competitive and profitable nature.

Extractive Question Answering extracts a span of text from a given context section as the answer to a specified query. With the introduction of sizeable pre-trained language models, such as BERT [3], which employ the Transformers [4] architecture to design robust language models for various NLP tasks determined by benchmarks, such as GLUE [5] or decaNLP [6], the Question Answering system has seen considerable progress. But because of the introduction of new datasets, NewsQA and SubjQA, that rely considerably on reasoning, it becomes contesting to generalize prior performing QA models to various datasets. The work done by us analyzes the contrast between various pre-trained transformer-based language models to examine how well they can generalize to datasets of varying levels of complicatedness when fine-tuned on the question-answering task. Furthermore, we propose a new architecture by combining the idea of Dynamic Masking and Stride Shift to handle long-form subjective and opinionated answers and evaluate its performance against traditional pre-trained models on extractive question answering tasks.

Closed Domain Question Answering system is an intelligent system that answers a user's query. It comes under the field of Information Retrieval tasks. The key feature of the proposed solution is the ability to answer non-factoid-based questions in a human behavioral manner. It will first find the relevant articles and then identify the answer span of those articles. The modular Extractive Question Answering System comprises two components: Firstly, it should rank the pertinent articles of a knowledge base (like Wikipedia). Secondly, it should extract answers from the various relevant articles retrieved by the ranker.

The architecture of the proposed solution is shown in Fig. 1.

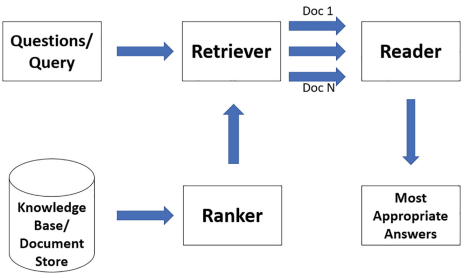

Fig. 1. Non-Factoid Question Answering System Architecture

The design of the remaining paper is as follows: First, there's a discussion regarding the relevant literature in the financial question answering system analysis and pre-trained linguistic models (Sect. 2). Then, there's an illustration of

the evaluated models (Sect. 3). After that, there's a description of the experimental setup used (Sect. 4). In Sect. 5, we have presented our experimental results on the financial question answering datasets (FiQA) and SubjQA [7] datasets. Finally, we discussed the future work and then the conclusion.

2 Related Literature

Earlier, many researchers adopted various approaches to develop a natural language question-answering system in the financial domain. Still, the focus of their work is around rule-based, and word counting system approaches [2,3,8]. Although they have good explainability power, these systems cannot capture the longer dependencies and context in a corpus [1]. Moreover, the task of QA becomes more difficult when the answers are subjective and opinionated, as in the case of the SubjQA dataset. Here the answers to a given query depend on the personal experience of the users, which makes the task potentially more difficult than finding answers to factual questions.

John M. Boyer [1] first developed a binary question classifier using the Naive Bayes algorithm to determine whether a question is a financial perspective or an informational question based on the number of financial keywords. However, financial entities such as currency, assets, and industry are more common, leading to biases and misclassifications of domain terms. A rules-based system was developed in which domain terms were selected and substituted to address the previous issue. The informational questions were marked as non-factoid questions for which logistic regression was used, operating over 80 proprietary linguistic scorers. The remaining perspective questions were marked as factoid questions.

Wen-tau Yih et al. [9] suggested using the WordNet lexical database to map semantically related words and find similarities between questions and answers. But since the previous two approaches were based on feature engineering methodologies and linguistic matching, they could not represent a domain-specific financial language.

In a non-factoid Question Answering system, each question has to be compared with a pool of answer candidates to determine a relevance rating; using the entire answer space as a candidate pool would be ineffective. Nam Khanh Tran et al. [10] proposed a deep learning framework for answering non-factoid-based questions. Moreover, their implementation reduced the answer space by incorporating a non-machine learning answer retrieval system, which focuses on ranking answers that are likely to be relevant. They introduced two main components called Answer Retriever and Answer Ranker.

Many representation learning techniques have been implemented previously to get the embedding vector of question and answer and to use these vectors to select the most relevant answers by matching the vector representations [11–13]. The Siamese architecture uses the same encoder network as the RNNs to individually create the embedding vector of the questions and answers. Although the same encoder is used, the calculations of the question and answer embedding representations are not impacted by one another, as they are positioned individually.

In 2017 Shuohang Wang et al. [13] proposed a Compare-Aggregate framework that initially compares all the words of a question and answers. After that, the results are aggregated by a Recurrent Network or Convolutional Neural Network into a vector embedding to compute the final relevance score. It can capture the contextual information more accurately than the previously proposed architecture like Siamese.

Nam Khanh Tran et al. [10] proposed a deep learning architecture called SRanker using the Siamese network architecture and Glove embeddings to implement QA systems in the financial domain. But instead of using the pooling layer in CNN (encoder network), they applied an attention mechanism to give more weight to the words that have more influence on the final representation. Nam Khanh Tran et al. modified the Compare-Aggregate architecture to construct the CARanker non-factiod question answering system in financial domain. The CARanker initially processes the questions and answers in the embedding layer. After that, an attention matrix is calculated based on the questions and answers embedding vectors in the attention layer. Each answer is then compared to a weighted question that reasonably matches the answer in the comparison layer using a comparison function. Finally, in the aggregation layer, the resultant embedding vector from the prior layer is aggregated using a 1-layer CNN network to calculate the conclusive score, which is used to order candidate responses.

The contributions of our works are as follows:

- We presented an end-to-end Extractive Question Answering using RoBERTa-base-squad2 combined with Stride Shift, Dynamic Masking, and Dense Passage Retriever for review/opinionated-based datasets where the answer to a given question depends upon personal opinion, i.e., FiQA and SubjQA.
- It consists of three components: Document Store to store high volumes of data and quickly filter it with full-text search, Retriever to extract relevant documents for a given question, and Reader to examine every document and extract the most suitable answer from the records provided by the Retriever.
- The possibilities where an answer to a given question could lie near the end of the long passages, merely truncating the long texts under the presumption that embedding of start token <s> retains the sufficient knowledge is problematic. So we applied the Stride Shift strategy to decrease the spatial resolution while handling long contextual passages, which leads to computational benefits.
- We have re-implemented the Dynamic Masking for RoBERTa but with an increased masking rate. We generated the masking pattern for every epoch every time we fed a sequence to the model. But instead of using default 15% as the masking rate, We have experimented with various masking rates ranging from 10% to 40%. But we found that masking 35% of the tokens in every epoch makes the model more robust toward the opinionated data.

3 Method

This section introduces various Transformer-based Language Model implementations for the Question Answering System for the FiQA and SubjQA (Books Review) datasets.

3.1 Preliminaries

LSTM. Long Short Term Memory is a type of RNN network that avoids vanishing gradients. Moreover, it allows long-term dependencies in a sequence to endure in the network by using "forget" and "update" gates. In LSTM at each time step, each word of a sentence is taken as input.

ELMo. ELMo (Embeddings from Language Model) uses a bi-directional LSTM, which is another version of an RNN and you have the inputs from the left and the right. First, the model is pre-trained with unlabeled data to get the embedding vector for each word. The language model weights are then determined and added to a specific task model for the supervised retraining, also known as a fine-tuning step. Although it is bi-directional, it suffered from some issues such as capturing longer-term dependencies.

Transformers. Transformers are attention-based networks with many encoder and decoder models, which are used for modeling sequential information. In a transformer, all the sentence words are taken as input simultaneously. The key idea behind transformers is the concept of self-attention (i.e., paying attention to other words in the same sentence). The encoder network consists of the multi-headed self-attention layer, which is used to compute the key, query, and value mappings from the embedding representation of the words. A similarity score is now computed by taking the dot product of each token's key and all tokens' query vectors used to generate the new representation for each token. Finally, these layers are concatenated together so that the sequence can be evaluated from varying "perspectives." After this, the resultant embedding representation will be passed through the fully connected feed-forward network.

BERT. BERT (Bi-directional Encoder Representation from Transformers) can be described as a sentence embedding model. There is no decoder in BERT. There is no concept of timesteps in BERT because at any point all the input can be seen from both the directions (right to left and left to right). The BERT architecture comprises two steps: pre-training and fine-tuning. The BERT model is trained on unlabelled data in various pre-training tasks during the pre-training. The BERT model is first initialized with the pre-trained parameters for fine-tuning, and all parameters are fine-tuned using selected data from subsequent tasks.

The input embeddings are the total of the token embeddings, the segmentation embeddings, and the position embeddings.

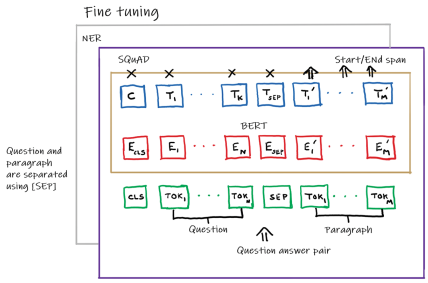

Fig. 2. Question Answering system using BERT

The question answering system using BERT is shown in Fig. 2. The C token in the image above could be used for classification purposes. The unlabeled sentence A/B pair will depend on what you are trying to predict, it could range from question answering to sentiment. (in which case the second sentence could be just empty). The BERT objective is defined as follows:

- Objective 1: Multi Mask Language Model where the loss is Cross entropy loss
- Objective 2: Next Sentence Prediction where the loss is Binary Loss.

RoBERTa. RoBERTa stands for "Robustly Optimized BERT pre-training Approach" [14]. RoBERTa is an improved technique for training BERT to improve performance. Firstly, the Next Sentence Prediction task is not useful for pre-training the BERT model. Therefore, the RoBERTa drops that as part of the objective here, which simplifies the presentation of examples (as in BERT, inputs are two concatenated document segments, whereas, in RoBERTa, inputs are sentence sequences that may span document boundaries) and the modeling objective. So RoBERTa is using a Masked Language Modeling objective. There are also changes in the size of the training batches. So for BERT, batch size was 256 examples, whereas RoBERTa increased it up to 2000. RoBERTa uses dynamic masking, wherein for different epochs, different parts of the sentences are masked, making the model more robust. There are some modifications to the process of tokenization as well. BERT uses a word piece tokenization approach that mixes sub-word pieces with whole words, whereas RoBERTa simplifies that down to just character-level byte-pair encoding.

4 Experimental Setup

4.1 Datasets

FiQA. We have used opinion-based Financial Question Answering datasets from task 2 of the FiQA challenge. The datasets consist of 6648questions in total, divided into the train, validation, and test set and 57640 answer passages with 17,110 QA pairs [15]. The training set consists of 5683 questions, and validation

set consists of 632 and test set consists of 333 questions. Each sample in the above three groups is a list of triples with a question id, ground truth answer ids, and a list of negative answer id.

SubjQA. SubjQA is an English Question Answer dataset containing more than 10000 customer reviews about products and services, which spans six different domains: Books, Electronics, Grocery, Restaurants, TripAdvisor, and Movies. It is a question-answering dataset that contains subjectivity labels for both questions and answers; they depend upon the customers' personal experiences. Here in our work, we have focused on building a QA system for the Books domain. The dataset consists of 1314 training examples and 256 validation examples. We have tested our model on 345 test examples. Each instance in the above three groups consists of 5 different attributes: *question_text*, *product_id*, *answer_text*, *answer_start*, and *passage*.

4.2 Baseline Model

Firstly we have implemented an LSTM classifier with GLoVe embeddings. We have used LSTM with the hidden dimension of 256, with the last hidden state size being 512 due to bidirectionality and a maximum sequence length of 128. A shared Bi-directional LSTM has been used as an encoder to generate the embedding vector of 100-dim for both question and answer independently (pre-trained GloVe embeddings is used for initializing embedding layer with a dimension of 100). Then a pooling layer has been used to generate a one embedding vector for both questions and answers. Bi-directional LSTM outputs one word at each time step. To avoid the overfitting of the network, we have applied dropout with a dropout rate = 0.2. Finally, the question and answer embedding vectors are compared using cosine similarity to get the best possible response. We trained our network using mini-batch SGD for three epochs with a batch size of 64 and a learning rate of 1e−3. Hinge loss has been used as a loss function. We have used another model, LSTM with ELMo embeddings, but the architecture is the same as before.

4.3 Evaluation Metrics

For the evaluation of the Question Answering system on FiQA dataset we have used 3 metrics: Mean Reciprocal Rank (MRR) [16] which is basically the mean of the Reciprocal Rank across multiple queries. The RR is defined as $\frac{1}{k}$ where k is the rank position of the first relevant ground truth answer.

$$MRR = \frac{1}{|Q|} \sum_{i=1}^{|Q|} \frac{1}{rank_i} \tag{1}$$

Normalized Discounted Cumulative Gain (NDCG) [16] is a normalized score which ensures that a more relevant document is discounted if it has a lower

rank.

$$DCG@k = rel_1 + \sum_{i=2}^{k} \frac{rel_i}{log_2(i+1)} \tag{2}$$

where k is the top k retrieved documents and rel_i is the relevance score at position i. Precision@1 determines the percentage of retrieved documents that are relevant to the query at the top 1 position [16].

To evaluate the performance of our fine-tuned models on SubjQA of the datasets, we have used the F1-score and Exact Match. F1-score is the harmonic mean of recall and precision, calculated for both the classified start token and the end token and averaged to get a single F1-score.

$$F1 - score = 2 * \frac{precision * recall}{precision + recall} \tag{3}$$

4.4 Implementation Details

We have experimented with the RoBERTa-base-squad2 model to implement a non-factoid question answering system. Since we do not want to train the RoBERTa model from scratch, we use the transfer learning technique. There are three main advantages to transfer learning: Reduce training time, improve predictions, and use smaller datasets (like FiQA, which is much smaller than SQuAD).

Firstly we loaded a pre-trained RoBERTa model from the Hugging face and then preprocessed the data to get the tokenized inputs and outputs: "question: Q, context: C" as input and "A" as the target. When there is no answer to a question given a context, we have used the *s* token, a unique token used to represent the start of the sequence. Tokenizers can split a given string into substrings, resulting in a subtoken for each substring, creating misalignment between the list of dataset tags and the labels generated by the tokenizer. So we have aligned the start and end indices with the tokens associated with the target answer word. Finally, a tokenizer can truncate a very long sequence. So, when the start/end position of an answer is None, we have assumed that it was truncated and assigned the maximum length of the tokenizer to those positions. After that, we fine-tuned the RoBERTa model on the new task and input, the FiQA dataset. The model returns the two logits as output; start logit and end logit. To get the answer, we have computed the argmax over the start logits and end logits for each token and then sliced the answer span from the inputs. The logits model also returns the probability score for each answer (to handle multiple answer cases) obtained by taking the softmax over the logits.

To deal with long passages which contain more than 512 tokens, we have used the stride shift method (similar to computer vision), where every window has been assigned a fixed passage of tokens that fits the model context's size. Then strides are shifted to give the subsequent set tokens to another window. Also, to introduce variability in the model, instead of using a similar mask for every input token in every iteration, RoBERTa applies the dynamic masking function

where the masks are generated in every iteration whenever an input sequence is passed to the model. But we found that masking 35% of the tokens in every epoch makes the model more robust toward the opinionated data.

We trained our network for three epochs with a batch size of 4 and a learning rate of 1e−05 with a weight decay of 0.01. The Adam optimizer with adaptive learning rates has optimized the cross-entropy loss function. To build an end-to-end QA pipeline, we have used Retriever-Reader architecture.

Retriever. Retriever is a lightweight filter responsible for extracting relevant documents for a given question by scanning all the documents in the Document store and identifying the suitable candidate set of documents. To achieve a good performance result, we have used a Dense Passage Retriever which uses encoders like transformers to represent the query and document as dense embedding vectors. These vectors encode semantic meaning and allow the dense retrievers to improve search accuracy by understanding the context of the question.

Reader. Reader examines every document and extracts the most suitable answer from the records provided by the retriever. We have used Deepset's FARM Reader for fine-tuning and deploying our Language Models. We can also perform an inter-passage answer comparison using FARM Readers, and the logits are not normalized. Moreover, it also removes the duplicate answers.

5 Results

The results of the baseline model LSTM (with glove embeddings), LSTM with ELMo, BERT-base-uncased model fine-tuned on the Financial dataset for the Question Answering task and RoBERTa-base using Stride shift and Dynamic Masking which is fine-tuned on FiQA dataset can be seen in Table 1.

Table 1. Experimental Results on the Financial Question Answering dataset (FiQA)

Model	Performance Metrics				
	Loss	Accuracy	MRR	NDCG	Precision@1
$SRanker_{mlp}$			0.242	0.278	0.119
CARanker			0.279	0.308	0.157
LSTM	0.366	76.48	0.136	0.096	0.036
LSTM with ELMo	0.219	83.68	0.098	0.143	0.054
BERT-base-Uncased	0.11	87.42	0.354	0.418	0.317
RoBERTa-base with stride shift	0.08	89.63	0.458	0.419	0.372

RoBERTa-base using Stride Shift, Dynamic Masking and Dense Passage Retriever which is fine-tuned on FiQA dataset outperforms all other models we implemented (LSTM, LSTM with ELMo, BERT) and the models reported by other papers for all measured metrics. LSTM with ELMo embeddings is better than LSTM with static embeddings like GloVe in all metrics except MRR.

Table 2 shows F1-score and Exact Match of various pre-trained Models on SubjQA (Books Domain) dataset.

Table 2. F1-scores and EM for various pre-trained models on SubjQA (Books Domain) Datasets

Model	Performance Metrics	
	F1-Score	EM
$BERT_{BASE}$	0.6220	0.6355
$RoBERTa_{BASE}$	0.6331	0.6395
XLM-RoBERTa	0.6562	0.6405
$RoBERTa_{BASE}$ with Stride Shift and DPR	0.6669	0.6484

The RoBERTa-base model using Stride Shift, Dynamic Masking and Dense Passage Retriever is compared with baselines in Table 2. The results show that the idea of using Stride Shift and Dense Passage Retriever improves the performance of answer-selection models. The F1-score and Exact Match (EM) metrics are increased for SubjQA datasets. In this model, the F1-score and EM metrics are improved by 1.016% and 1.012%, respectively.

Transformer-based Language Models that are fine-tuned on SQuAD will usually generalize satisfactorily to other domains. But for FiQA, we have observed that the MRR (Mean Reciprocal Rank), NDCG(Normalized Discounted Cumulative Gain), and Precision@1 of our model were considerably poorer than for SQuAD. This failure to generalize has also been marked in different review/opinion based datasets like SubjQA and is comprehended as proof that transformer-based language models are notably adept at overfitting to SQuAD datasets.

6 Conclusions and Future Work

We have applied a fine-tuned RoBERTa model to the financial question answering dataset (FiQA) in this project and combined it with stride shift methodology to handle long-form answers and Dense Passage Retriever technique to prevent the model from returning duplicate answers. Pre-trained RoBERTa models enabled us to mitigate the disadvantages of low data density, the specificity of financial language, and the external use of pre-trained dynamic word embeddings from conventional deep learning methods. This paper compares the performance

of between different pre-trained language models fine-tuned on question answering datasets of varying difficulty levels. Exploratory results show the effectiveness of our approaches and demonstrate that RoBERTa-base combined with Stride Shift methodology, Dynamic Masking, and Dense Passage Retriever improve model performance in question answering. We observe at least 3% increase in MRR and Precision@1 performance metrics over the BERT base model on the FiQA dataset.

Although we were getting good results on most of the questions of test data, we still found some scope for fine-tuning. In future we plan to carry out this fine tuning. Because of fewer data, training the Question Answering System on Synthetic data will help to build a more robust model. In this paper, we have only extracted answer spans from the context/passage. Still, in general, it could be that bits and pieces of the answer are sprayed throughout the document, and we would like our model to synthesize these components into a single legible response. Moreover, most existing solutions rely on the answer-span in the text corpora, but what if i) it is not present, or ii) wrongly annotated. To handle these cases, we can generate answers as the span of text in a document using a pre-trained language model and produce better-phrased answers that synthesize evidence across multiple passages.

References

1. Boyer, J.M.: Natural language question answering in the financial domain. In: Onut, I.V., Jaramillo, A., Jourdan, G.-V., Petriu, D.C., Chen, W., (eds.) Proceedings of the 28th Annual International Conference on Computer Science and Software Engineering, CASCON 2018, Markham, Ontario, Canada, 29–31 October 2018, pp. 189–200. ACM (2018)
2. Araci, D.: FinBERT: financial sentiment analysis with pre-trained language models. CoRR, abs/1908.10063 (2019)
3. Devlin, J., Chang, M. W., Lee, K., Toutanova, K.: BERT: pre-training of deep bidirectional transformers for language understanding. In: Burstein, J., Doran, C., Solorio, T. (eds.) Proceedings of the 2019 Conference of the North American Chapter of the Association for Computational Linguistics: Human Language Technologies, NAACL-HLT 2019, Minneapolis, MN, USA, 2–7 June 2019, vol. 1 (Long and Short Papers), pp. 4171–4186. Association for Computational Linguistics (2019)
4. Vaswani, A., et al.: Attention is all you need. In: Guyon, I., et al., (eds.) Advances in Neural Information Processing Systems 30: Annual Conference on Neural Information Processing Systems 2017, 4–9 December 2017, Long Beach, CA, USA, pp. 5998–6008 (2017)
5. Wang, A., Singh, A., Michael, J., Hill, F., Levy, O., Bowman, S.R.: GLUE: a multitask benchmark and analysis platform for natural language understanding. In: 7th International Conference on Learning Representations, ICLR 2019, New Orleans, LA, USA, 6–9 May 2019. OpenReview.net (2019)
6. McCann, B., Keskar, N.S., Xiong, C., Socher, R.: The natural language decathlon: multitask learning as question answering. CoRR, abs/1806.08730 (2018)

7. Bjerva, J., Bhutani, N., Golshan, B., Tan, W.C., Augenstein, I.: SubjQA: a dataset for subjectivity and review comprehension. In: Webber, B., Cohn, T., He, Y., Liu, Y., (eds.) Proceedings of the 2020 Conference on Empirical Methods in Natural Language Processing, EMNLP 2020, Online, 16–20 November 2020, pp. 5480–5494. Association for Computational Linguistics (2020)
8. Feng, G., et al.: Question classification by approximating semantics. In: Gangemi, A., Leonardi, S., Panconesi, A., (eds.) Proceedings of the 24th International Conference on World Wide Web Companion, WWW 2015, Florence, Italy, 18–22 May 2015 - Companion Volume, pp. 407–417. ACM (2015)
9. Yih, S.W.T., Chang, M.-W., Meek, C., Pastusiak, A.: Question answering using enhanced lexical semantic models. In: Proceedings of the 51st Annual Meeting of the Association for Computational Linguistics, ACL 2013, 4–9 August 2013, Sofia, Bulgaria, vol. 1: Long Papers, pp. 1744–1753. The Association for Computer Linguistics (2013)
10. Tran, N.K., et al.: A neural network-based framework for non-factoid question answering. In: Champin, P.A., Gandon, F., Lalmas, M., Ipeirotis, P.G., (eds.) Companion of the the Web Conference 2018 on The Web Conference 2018, WWW 2018, Lyon, France, 23–27 April 2018, pp. 1979–1983. ACM (2018)
11. Feng, M., Xiang, B., Glass, M. R., Wang, L., Zhou, B.: Applying deep learning to answer selection: a study and an open task. In: 2015 IEEE Workshop on Automatic Speech Recognition and Understanding, ASRU 2015, Scottsdale, AZ, USA, 13–17 December 2015, pp. 813–820. IEEE (2015)
12. Tan, M., Xiang, B., Zhou, B.: LSTM-based deep learning models for non-factoid answer selection. CoRR, abs/1511.04108 (2015)
13. Wang, S., Jiang, J.: A compare-aggregate model for matching text sequences. In: 5th International Conference on Learning Representations, ICLR 2017, Toulon, France, 24–26 April 2017, Conference Track Proceedings. Open-Review.net (2017)
14. Liu, Y.: RoBERTa: a robustly optimized BERT pretraining approach. CoRR, abs/1907.11692 (2019)
15. Maia, M., et al.: Www'18 open challenge: financial opinion mining and question answering. In: Champin, P.-A., Gandon, F., Lalmas, M., Ipeirotis, P.G., (eds.) Companion of the the Web Conference 2018 on The Web Conference 2018, WWW 2018, Lyon, France, 23–27 April 2018, pp. 1941–1942. ACM (2018)
16. Mogotsi, I.C., Manning, C.D., Raghavan, P., Schütze, H.: Introduction to information retrieval. Inf. Retr. **13**(2), 192–195 (2010). Cambridge University Press, Cambridge, England, 2008, pp. 482, ISBN 978-0-521-86571-5