# Improving Biomedical Question Answering with Sentence-based Ranking at BioASQ-11b

Notebook for the BioASQ Lab at CLEF 2023

Anna Aksenova[1], Tsvetan Asamov[1], Petar Ivanov[1] and Svetla Boytcheva[1]

[1]*Ontotext, Bulgaria*

**Abstract**

The paper presents a solution of BioASQ 2023 11b question answering task (part of the Conference and Labs of the Evaluation Forum - CLEF). Our team participated in Phase B, submitting the system for `factoid` and `yesno` types of questions in English based on extractive question answering and text classification respectively. In this work, we outline our Question Answering (QA) approach based on sentence embedding ranking coupled with biomedical ELECTRA model [1] fine-tuning. The approach showed the third-best accuracy score for yesno (0.8571) questions and the fourth-best accuracy score for `factoid` questions (0.5161) on the final test set.

**Keywords**

ELECTRA, Extractive Question Answering, Sentence BERT, Information Retrieval, Biomedical NLP

## 1. Introduction

This paper describes a pipeline proposed by our team for solving BioASQ-11b [2] Phase B task. We propose a simple yet efficient approach to biomedical question answering that may further be scaled and used for industrial purposes.

The BioASQ 2023 task 11b phase B consisted of several subtypes of questions in English to be resolved, namely `factoid`, `yesno` or `list` questions. For each of the types, the participants were encouraged to provide not only an "exact answer" (span of text, binary value and list of entities respectively), but also a comprehensive "ideal answer", which would answer the question in a natural way. Our team focused on retrieving answers for `factoid` and `yesno` questions also providing ideal answers corresponding to those tasks.

In this work, we present our approach which is based on 2 steps: sentence embedding cosine similarity ranking based on biomedical sentence BERT [3] and a fine-tuning BioM-ELECTRA model [1] on text and token classification.

The paper is organized as follows: in Section 2 we present a brief outline of the previous research on question answering task; Section 3 introduces task description; Section 6 describes our approach and experimental set up; in Section 4 we overview the challenge dataset and

*CLEF 2023: Conference and Labs of the Evaluation Forum, September 18–21, 2023, Thessaloniki, Greece*

anna.aksenova@ontotext.com (A. Aksenova); tsvetan.asamov@ontotext.com (T. Asamov); petar.ivanov@ontotext.com (P. Ivanov); svetla.boytcheva@ontotext.com (S. Boytcheva)

0000-0002-3489-874X (A. Aksenova); 0000-0002-7556-1350 (T. Asamov); 0000-0001-8448-1005 (P. Ivanov); 0000-0002-5542-9168 (S. Boytcheva)

additional sources used for model fine-tuning; in Section 5 the submitted system is explained, Section 7 discusses our main findings; finally, Section 8 draws some conclusions and outlooks for future work.

## 2. Related work

Early research in extractive QA focused on knowledge-based approaches that relied on structured data sources, such as biomedical ontologies and databases. Systems like the first BioASQ solutions [4] and AskHERMES [5] employed a combination of information retrieval techniques and named entity recognition to extract relevant information from curated resources. These approaches provided accurate answers by leveraging domain-specific knowledge, but their performance was limited by the availability and coverage of structured resources.

To overcome the limitations of knowledge-based approaches, researchers explored corpus-based approaches that utilized large-scale text corpora. Extractive QA in this setting has been solved quite well for the common domain after the introduction of the SQuAD dataset [6]. The top-ranked solutions rely on transformer-based models such as BERT [7] and XLNet [8]. These approaches leveraged the abundance of textual data, enabling broader coverage and adaptability to different question types.

As for the biomedical domain, the number of pre-trained language models of different architectures is quite limited. Biomedical transformer models follow the core architecture of the original transformer model, consisting of multi-head self-attention mechanisms and feed-forward neural networks. However, several model variants have been introduced to enhance their performance in the biomedical domain. For instance, models like BioBERT [9] and SciBERT [10] are transformer models pre-trained on biomedical text, providing domain-specific embeddings. Other variants include BlueBERT [11], ClinicalBERT [12], and PubMedBERT [13], which cater to specific subdomains or incorporate additional contextual information. These models provide a robust foundation for processing biomedical text and extracting valuable information from vast amounts of biomedical literature. According to the recent results, ELECTRA model [14] pre-trained on PubMed and fine-tuned on SQuAD showed state-of-the-art results on BioASQ-7 for base-scale models [15], therefore we focused on ELECTRA-based models for tackling BioASQ-11 challenge.

Providing a comprehensive and elaborated answer to a question could be approached as a natural language generation task. Leveraging the power of GPT-based architectures and pre-training techniques, BioGPT [16] has been developed to be applied for language modelling in the specific domain. BioGPT benefits from its pre-training on biomedical literature, which equips it with a strong foundation of domain-specific knowledge. It can understand and handle the technical terminology, abbreviations, and concepts prevalent in the biomedical field. This specialized knowledge allows BioGPT to effectively tackle complex biomedical questions that may require a deep understanding of the domain, enabling it to provide accurate and informative answers.

## 3. Task formulation

For each of the subtasks that we tackled, each data point consisted of a number abstracts extracted from PubMed scientific medical publications in English [1] and a question. The answer was supposed to be inferred from one or more of the given paragraphs.

- `Factoid` question answering
  Given a set of text paragraphs and an open question, return a short span of text containing the entity. Usually, it is a symptom, disease or numerical value.
- `Yesno` question answering
  Given a set of text paragraphs and a general question, return either "yes" or "no".
- Ideal answers formulation
  Given a set of text paragraphs and a general question, return a sentence in a natural language that will answer the question.

## 4. Data

The BioASQ dataset [17] was manually annotated for Question Answering (QA) task by medical experts. Originally the training set for `factoid` questions consisted of 1417 samples and the training set for `yesno` questions consisted of 1271 samples.

For the purposes of building different experimental systems, we have introduced several adjustments for those sets.

1. For internal evaluation purposes, original BioASQ-11b training datasets were split into train and development subsets, which comprised 80% and 20% of the full challenge data respectively.
2. For the `factoid` question set we checked whether the exact answer (or its non-capitalized version) was present in any of the reference abstracts. If so, we kept the question and extracted the position of the answer in the text, otherwise, we omitted the sample. [2] As a result, we obtained a dataset in standard SQuAD format [6].
3. For the `factoid` question training set we also employed additional training data based on BioASQ 7b. The dataset was transformed to SQuAD format and introduced by Jeong et al. [18]. The original dataset comprised 3231 questions. Duplicate questions were omitted. This particular dataset was used as, to the extent of our knowledge, it is the only open source biomedical QA dataset that has required SQuAD formatting.
4. For the `yesno` QA task we experimented on adopting PubMedQA [19] data to enlarge the training set. It included 1000 questions.
   As a result we obtained several train sets. The data distribution is presented in Table 1.

[1] https://pubmed.ncbi.nlm.nih.gov/

[2] Circa 30% of the datapoints did not contain exact answer that was expected.

**Table 1**
Data distribution for the obtained train and development sets

| Data | Train examples | Dev examples |
|---|---|---|
| Factoid cleaned | 892 | 222 |
| Factoid cleaned + BioASQ7 | 2891 | 222 |
| Yesno | 889 | 382 |
| Yesno + PubMedQA | 1889 | 382 |

# 5. Methodology

## 5.1. Factoid answers

As it was already mentioned in Section 4, we decided to narrow down the `factoid` QA task to extractive question answering, i.e. we assume that the exact answer is explicitly present in at least one of the reference paragraphs. In short, the backbone model is fine-tuned for token classification task and predicts whether each token holds `answer_start`, `answer_end` or `other` position. Such approach is unable to predict the answers that are not explicitly present in the context, however such QA systems are much easier to train and control. As it was already discussed in Section 2, ELECTRA-based models proved to be efficient for biomedical QA, therefore we use those as a backbone for our further fine-tuning experiments. Some of the reference paragraphs in the training set are longer than the input sequence length for transformer model that we used. To avoid loosing the information, we split such examples into parts with an overlap of 128 tokens following the approach suggested by Huggingface Transformers tutorial [20]. In addition, we adapted `answer_start` and `answer_end` candidate ranking procedure suggested by the same resource.

## 5.2. Yesno answers

The binary type of questions was tackled as a binary classification task with the same backbone transformer model. Question and context separated by `[SEP]` token were fed into the transformer with a linear layer on top of the pooled output.

## 5.3. Ideal answers

We focused on fine-tuning open-source BioGPT model [16] for generating ideal answers.

We have performed BioGPT fine-tuning in prefix-tuning setting by introducing additional tokens `[QUESTION]`, `[CONTEXT]`, and `[ANSWER]`, which should prompt the model to generate answers after the input questions and contexts. The schema of the training input is presented below.

`[BOS] [QUESTION]` *Question text* `[CONTEXT]` *Context text* `[ANSWER]` *Answer text* `[EOS]`

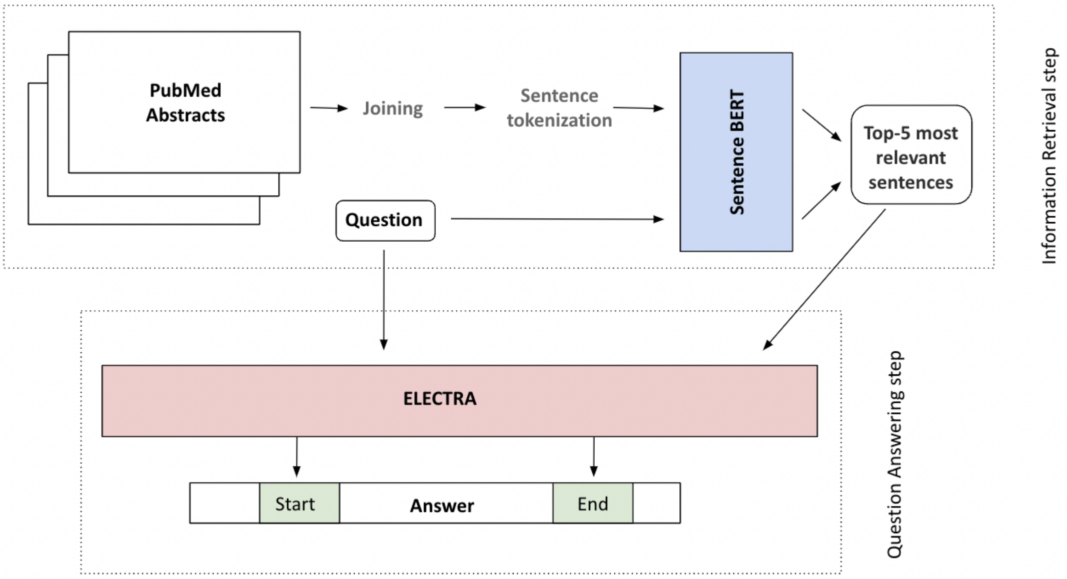

**Figure 1:** Proposed pipeline for `factoid` QA

## 5.4. Ranking

Typical end-to-end question answering pipelines include document retrieval step before the information extraction itself. We adopted this idea to the competition setting.

As the majority of answers in BioASQ can be given based on a single sentence from all of the given PubMed paragraphs, we decided to introduce sentence ranking step before QA.

Basic information retrieval pipeline consists of three steps: calculating document and query vectors, calculating similarity between query and documents, sorting the documents based on their similarity to the query. As we use sentences as items for ranking, for preprocessing sentence tokenization step we used SciSpacy small scientific model [21] as both rule-based and NLTK-based sentence splitting [22] were not able to preserve entities that the final QA system was supposed to extract. For instance, some abbreviations and measures with dots were identified as sentence borders. For embedding extraction we used sentence transformers [23] which are widely used as a basic architecture for ranking tasks. As such models are trained to increase cosine similarity between semantically close sentences, we sort all the sentences in reference passages by cosine similarity between sentence and question vector. Circa 96% of the answers were located in top-5 ranked sentences, therefore after ranking we reduce the contexts to top-5 sentences. Apparently, not only does this procedure reduce required training resources, but also improves the accuracy of the whole pipeline. Figure 1 presents all steps in the pipeline of the best submitted system for `factoid` questions.

### 5.5. Backbone models

Four publicly available biomedical transformer models were used in our experiments, namely BioGPT [16], BioM-ELECTRA [15], ELECTRAMed [24], S-PubMedBERT [3].

- *BioGPT*[3]: GPT-2-based model pre-trained on PubMed abstracts with custom-built vocabulary.
- *BioM-ELECTRA*[4]: pre-trained on PubMed Abstracts with PubMedBERT vocabulary [25] and fine-tuned on SQuAD 2.0.
- *ELECTRAMed*[5]: pre-trained on PubMed Abstracts with SciVocab vocabulary [10].
- *S-PubMedBERT* [6]: initialised as PubMedBERT and fine-tuned on MS-MARCO dataset [26] using sentence-transformers framework.

## 6. Experiments

The performance of our methods is reported on development set that we described in Section 4. Evaluation for all the tested systems is done with the target metrics of BioASQ challenge, i.e. accuracy for `yesno` questions, strict accuracy or exact match for `factoid` questions and ROUGE [27] for ideal answers.

All the pipelines were implemented using HuggingFace Transformers (Extractive QA, Text Classification and Language Modeling tutorials).

Table 2 and Table 3 report our experimental results. Initially, for obtaining exact answers, we fine-tuned both ELECTRA models without any updates on the data to establish a solid baseline. Then we experimented with adding more samples to the training sets for both types of questions. In particular, 2753 questions were added to `factoid` dataset and 1000 questions were added to `yesno` dataset. The models fine-tuned on the updated datasets are marked with "+" sign. Given that additional data improved the performance of extractive QA system, but did not help in terms of yesno QA, we took the best datasets for further experiments with ranking.

All the experiments were conducted on a single NVIDIA RTX A5000 GPU. For fine-tuning the models we used the following hyper parameter settings:

- `learning rate`: Initialized to 1e-5 and 5e-5. The latter is suggested as the best rate in the original BioM-ELECTRA paper by Alrowili and Vijay-Shanker [1]. The smaller learning rate appeared to be more beneficial in our case.
- `number of epochs`: We tested 3, 5 and 10 epochs for each of the settings. The best performance was achieved on 5 epochs, therefore we report these results.
- `batch size`: It was set to 16 due to the limitations of GPU memory.

As we were mostly focusing on exact answers, the number of experiments performed on ideal answer generation was limited. They are presented in Table 4. The model was fine-tuned on language modeling task with the following hyper parameters:

[3] https://huggingface.co/microsoft/biogpt

[4] https://huggingface.co/sultan/BioM-ELECTRA-Large-SQuAD2

[5] https://huggingface.co/giacomomiolo/electramed_base_scivocab_1M

[6] https://huggingface.co/pritamdeka/S-PubMedBert-MS-MARCO

**Table 2**
Evaluation of the tested approaches for `factoid` QA on dev set

| Pipeline | Strict accuracy | F1 |
|---|---|---|
| BioM-ELECTRA-SQuAD | 85.3 | 88.6 |
| BioM-ELECTRA-SQuAD+ | 87.0 | 90.1 |
| BioM-ELECTRA-SQuAD+_ranking | **88.5** | **91.9** |
| ELECTRAMed | 83.0 | 86.4 |
| ELECTRAMed+ | 84.9 | 88.9 |
| ELECTRAMed+_ranking | 86.7 | 90.1 |

**Table 3**
Evaluation of the tested approaches for `yesno` QA on dev set

| Pipeline | Accuracy |
|---|---|
| BioM-ELECTRA-SQuAD | 0.94 |
| BioM-ELECTRA-SQuAD+ | 0.92 |
| BioM-ELECTRA-SQuAD_ranking | **0.96** |
| ELECTRAMed | 0.90 |
| ELECTRAMed+ | 0.85 |
| ELECTRAMed_ranking | 0.92 |

- `learning rate`: Initialized to 1e-5 as a default suggested for the fine-tuning.
- `number of epochs`: We tested 3 and 5 epochs for each of the settings. As the models after 5 epochs fine-tuning provided better performance, we report the metrics for those.
- `batch size`: It was set to 4 due to the limitations of GPU memory.
- `generation temperature`: Initialized to 0.7 as a default parameter. We wanted our model to have some variability in generation, however we did not aim to allow it to generate very creative responses.
- `maximum prediction length`: Initialized to 100 tokens as an approximation of ideal answers in the training set.

**Table 4**
Evaluation of BioGPT for `ideal answer` generation

| Model | ROUGE |
|---|---|
| BioGPT | **0.42** |
| BioGPT_ranking | 0.40 |

## 7. Discussion

In terms of obtaining exact answers, re-ranking the context sentences improved the performance of both models for both types of questions. Between the two ELECTRA models, BioM-ELECTRA-

SQuAD showed slightly better results.

As for the ideal answers, limiting the context with sentence-ranking technique did not improve the target metric. Further research should be conducted in that direction.

The proposed system for exact answers presented second-best accuracy score on Test batch 3 leaderboard and third-best accuracy score on Test batch 4. The difference in scores on our development set and on the leaderboard is caused by a set of answers that is not explicitly present in any of the reference PubMed abstracts as the system cannot predict those by design. Although our system is inferior of top-1 GPT-3.5-based pipeline for `factoid` questions due to this limitation, it scores the same as GPT-4-based system for the same set of questions. In addition, our pipeline scores higher in lenient accuracy (top-5 accuracy), meaning that it is more beneficial for medical suggestions systems. Overall, the pipeline is quite flexible and not demanding in terms of computational resources. It can be easily customised for different datasets, domains and languages as soon as there exist relevant backbone transformer models.

## 8. Conclusion

In this paper we have presented simple yet efficient pipeline for building a question-answering system for biomedical domain based on pre-trained biomedical transformer models. The system showed good results in BioASQ 2023, proving that it could be used for further development and could be adapted to real-world medical question answering tasks.

As a direction for future development we can suggest focusing more on answer candidate selection step as the top-5 accuracy of our system is 20% higher compared to top-1 accuracy. In addition, we are aiming to scale this approach to multilingual question answering task as multilinguality still remains a challenge for QA systems, especially in biomedical domain.

## References

[1] S. Alrowili, K. Vijay-Shanker, Biom-transformers: building large biomedical language models with bert, albert and electra, in: Proceedings of the 20th Workshop on Biomedical Language Processing, 2021, pp. 221–227.

[2] A. Nentidis, A. Krithara, G. Paliouras, E. Farré-Maduell, S. Lima-López, M. Krallinger, Bioasq at clef2023: The eleventh edition of the large-scale biomedical semantic indexing and question answering challenge, in: Advances in Information Retrieval, Springer Nature Switzerland, Springer Nature Switzerland, Cham, 2023. URL: https://link.springer.com/chapter/10.1007/978-3-031-28241-6_66.

[3] P. DEKA, A. JUREK-LOUGHREY, P. DEEPAK, Improved methods to aid unsupervised evidence-based fact checking for online health news, Journal of Data Intelligence 3 (2022) 474–504.

[4] G. Balikas, A. Kosmopoulos, A. Krithara, G. Paliouras, I. Kakadiaris, Results of the bioasq tasks of the question answering lab at clef 2015, in: CLEF 2015, 2015.

[5] Y. Cao, F. Liu, P. Simpson, L. Antieau, A. Bennett, J. J. Cimino, J. Ely, H. Yu, Askhermes: An online question answering system for complex clinical questions, Journal of biomedical informatics 44 (2011) 277–288.

[6] P. Rajpurkar, J. Zhang, K. Lopyrev, P. Liang, Squad: 100,000+ questions for machine comprehension of text, arXiv preprint arXiv:1606.05250 (2016).

[7] J. Devlin, M.-W. Chang, K. Lee, K. Toutanova, Bert: Pre-training of deep bidirectional transformers for language understanding, 2019. arXiv:1810.04805.

[8] Z. Yang, Z. Dai, Y. Yang, J. Carbonell, R. Salakhutdinov, Q. V. Le, Xlnet: Generalized autoregressive pretraining for language understanding, 2020. arXiv:1906.08237.

[9] J. Lee, W. Yoon, S. Kim, D. Kim, S. Kim, C. H. So, J. Kang, Biobert: a pre-trained biomedical language representation model for biomedical text mining, Bioinformatics 36 (2020) 1234–1240.

[10] I. Beltagy, K. Lo, A. Cohan, Scibert: A pretrained language model for scientific text, arXiv preprint arXiv:1903.10676 (2019).

[11] Y. Peng, S. Yan, Z. Lu, Transfer learning in biomedical natural language processing: An evaluation of bert and elmo on ten benchmarking datasets, 2019. arXiv:1906.05474.

[12] K. Huang, J. Altosaar, R. Ranganath, Clinicalbert: Modeling clinical notes and predicting hospital readmission, arXiv preprint arXiv:1904.05342 (2019).

[13] Y. Gu, R. Tinn, H. Cheng, M. Lucas, N. Usuyama, X. Liu, T. Naumann, J. Gao, H. Poon, Domain-specific language model pretraining for biomedical natural language processing, ACM Transactions on Computing for Healthcare (HEALTH) 3 (2021) 1–23.

[14] K. Clark, M.-T. Luong, Q. V. Le, C. D. Manning, Electra: Pre-training text encoders as discriminators rather than generators, arXiv preprint arXiv:2003.10555 (2020).

[15] S. Alrowili, V. Shanker, BioM-transformers: Building large biomedical language models with BERT, ALBERT and ELECTRA, in: Proceedings of the 20th Workshop on Biomedical Language Processing, Association for Computational Linguistics, Online, 2021, pp. 221–227. URL: https://aclanthology.org/2021.bionlp-1.24. doi:10.18653/v1/2021.bionlp-1.24.

[16] R. Luo, L. Sun, Y. Xia, T. Qin, S. Zhang, H. Poon, T.-Y. Liu, BioGPT: generative pre-trained transformer for biomedical text generation and mining, Briefings in Bioinformatics 23 (2022). URL: https://doi.org/10.1093/bib/bbac409. doi:10.1093/bib/bbac409. arXiv:https://academic.oup.com/bib/article-pdf/23/6/bbac409/47144271/bbac409.pdf, bbac409.

[17] A. Krithara, A. Nentidis, K. Bougiatiotis, G. Paliouras, Bioasq-qa: A manually curated corpus for biomedical question answering, Scientific Data 10 (2023) 170. URL: https://doi.org/10.1038/s41597-023-02068-4.

[18] M. Jeong, M. Sung, G. Kim, D. Kim, W. Yoon, J. Yoo, J. Kang, Transferability of natural language inference to biomedical question answering, arXiv preprint arXiv:2007.00217 (2020).

[19] Q. Jin, B. Dhingra, Z. Liu, W. Cohen, X. Lu, Pubmedqa: A dataset for biomedical research question answering, in: Proceedings of the 2019 Conference on Empirical Methods in Natural Language Processing and the 9th International Joint Conference on Natural Language Processing (EMNLP-IJCNLP), 2019, pp. 2567–2577.

[20] T. Wolf, L. Debut, V. Sanh, J. Chaumond, C. Delangue, A. Moi, P. Cistac, T. Rault, R. Louf, M. Funtowicz, J. Davison, S. Shleifer, P. von Platen, C. Ma, Y. Jernite, J. Plu, C. Xu, T. Le Scao, S. Gugger, M. Drame, Q. Lhoest, A. Rush, Transformers: State-of-the-art natural language processing, in: Proceedings of the 2020 Conference on Empirical Methods in Natural Language Processing: System Demonstrations, Association for Computational

Linguistics, Online, 2020, pp. 38–45. URL: https://aclanthology.org/2020.emnlp-demos.6. doi:10.18653/v1/2020.emnlp-demos.6.

[21] M. Neumann, D. King, I. Beltagy, W. Ammar, ScispaCy: Fast and Robust Models for Biomedical Natural Language Processing, in: Proceedings of the 18th BioNLP Workshop and Shared Task, Association for Computational Linguistics, Florence, Italy, 2019, pp. 319–327. URL: https://www.aclweb.org/anthology/W19-5034. doi:10.18653/v1/W19-5034. arXiv:arXiv:1902.07669.

[22] S. Bird, E. Klein, E. Loper, Natural language processing with Python: analyzing text with the natural language toolkit, " O'Reilly Media, Inc.", 2009.

[23] N. Reimers, I. Gurevych, Sentence-bert: Sentence embeddings using siamese bert-networks, 2019. arXiv:1908.10084.

[24] G. Miolo, G. Mantoan, C. Orsenigo, Electramed: a new pre-trained language representation model for biomedical nlp, 2021. arXiv:2104.09585.

[25] Y. Gu, R. Tinn, H. Cheng, M. Lucas, N. Usuyama, X. Liu, T. Naumann, J. Gao, H. Poon, Domain-specific language model pretraining for biomedical natural language processing, ACM Transactions on Computing for Healthcare 3 (2021) 1–23. URL: https://doi.org/10.1145%2F3458754. doi:10.1145/3458754.

[26] P. Bajaj, D. Campos, N. Craswell, L. Deng, J. Gao, X. Liu, R. Majumder, A. McNamara, B. Mitra, T. Nguyen, M. Rosenberg, X. Song, A. Stoica, S. Tiwary, T. Wang, Ms marco: A human generated machine reading comprehension dataset, 2018. arXiv:1611.09268.

[27] C.-Y. Lin, ROUGE: A package for automatic evaluation of summaries, in: Text Summarization Branches Out, Association for Computational Linguistics, Barcelona, Spain, 2004, pp. 74–81. URL: https://www.aclweb.org/anthology/W04-1013.