# An Approach to Decentralized Hybrid Question Answering Systems

Dilan Bakır and Mehmet S. Aktas(✉)

Yildiz Technical University, Istanbul, Turkey
dilan.bakir@std.yildiz.edu.tr, aktas@yildiz.edu.tr

**Abstract.** Question answering (QA) systems are sequence-to-sequence model based programs that find the most accurate answer to a query by using given texts. In today's world, QA system software is run on different platforms by various institutions. This means that users have to use different QA system software, developed with different technologies on various platforms, to find the most accurate answer to their queries. However, there is a lack of systems that allow these types of systems to be used in a hybrid manner on a single interface. To address this issue, this research investigates a software architecture that will enable QA systems trained with different datasets on various platforms to be used with a single programming interface. To demonstrate the feasibility and usefulness of the proposed hybrid question answering system framework software architecture, a prototype software has been developed. The performance of the developed prototype software has been compared with standalone QA system software based on the execution time performance metric. The obtained results demonstrate that the proposed framework software architecture has negligible processing overheads.

**Keywords:** Hybrid System · Question answering · Artificial intelligence · Transformer · Sequence-to-Sequence Models

## 1 Introduction

In today's world, the vast amount of information available on the internet has made it increasingly difficult to find accurate answers to queries. To address this challenge, researchers have developed question answering systems that are capable of understanding natural language queries and providing relevant answers. However, these systems are often developed by different institutions and run on different platforms, making it difficult for users to access accurate answers in a timely manner.

In this context, the development of a hybrid question answering systems has emerged as a potential solution. The hybrid question and answer systems will enable users to access multiple question answering systems that have been trained on different data sets and run on different platforms through a single interface. This approach has the potential to significantly improve the efficiency and accuracy of finding information from large amounts of text data.

© The Author(s), under exclusive license to Springer Nature Switzerland AG 2023
O. Gervasi et al. (Eds.): ICCSA 2023 Workshops, LNCS 14112, pp. 29–38, 2023.
https://doi.org/10.1007/978-3-031-37129-5_3

This paper presents a study that investigates the feasibility of developing a hybrid question answering system framework. The proposed software architecture enables question answering systems to be used in a transparent manner by allowing systems trained on different data sets and run on different platforms to be accessed through a single interface. The authors developed a prototype software to demonstrate the usability of the proposed framework software architecture, and found that it is useful and feasible.

Overall, the results of this study have important implications for the future of hybrid question answering systems, as the development of hybrid systems could significantly improve the accessibility and accuracy of information retrieval from large amounts of text data.

In this study, we compare the results of T5 and BERT models in hybrid question answering systems. By integrating hybrid question answering systems, we aim to advance the field and improve the accuracy of natural language processing models.

In Sect. 1, provides an introduction to the study, highlighting the importance of hybrid question answering system. In Sect. 2, outlines the research questions we sought to answer through our study. In Sect. 3, provides a literature review of question answering systems fields, discussing the challenges and recent advancements in these areas. In Sect. 4, we give information about the methodology we propose. In Sect. 5, we cover the details of our prototype implementation and experimental work. In Sect. 6 describes the result of our work and how we should work for the future.

## 2 Research Questions

The aim of this research is to explore the feasibility of developing a hybrid question answering system framework that allows for the efficient and accurate retrieval of information from large amounts of text data. To achieve this aim, the study will focus on addressing the following research questions:

RQ1: What are the technical requirements and software architecture for developing a hybrid question answering system framework that can access multiple systems trained on different datasets and run on different platforms through a single interface?

RQ2: How can a hybrid question answering system framework be developed to enable multiple question answering systems to be accessed through a single interface?

RQ3: What are the key considerations for integrating question answering systems trained on different data sets and run on different platforms within a hybrid framework?

RQ4: How can we evaluate the performance of the hybrid Question Answering Systems from the execution time performance metric perspective?

By addressing these research questions, this study aims to provide insights into the development of hybrid question answering systems and their potential to

improve the accessibility and accuracy of information retrieval. The findings of this study could have important implications for the future development and application of question answering systems.

## 3 Question Answering Literature Review

Studies in the field of Question Answering first started as Knowledge Bases (KB). However, it has been observed that there are certain difficulties in extracting the answer from the raw text. With these difficulties, the need to be read and understood by a machine and the answer to be produced by the machine has increased. These need studies have led to Deep learning techniques. And with the development of deep learning, studies in the field of QA have increased.

Studies in the field of question Answering systems are increasing day by day. We have done a literature review [12] about Question Answering systems in our previous studies. When we examine the literature, the first studies were done on dynamic attention mechanisms [13], and context is updated with query weights [14]. Following this work, the memoryless Bidirectional+attention model (Bidaf) attention mechanism was developed [15]. Bidaf, one of the question Answering systems, is an extractive reader system that estimates the start and end position from the context and accepts that the answer is in the context. Transformer models such as BART, and T5 are generative systems that produce the answer randomly, not from within the context. Extractive readers have been found to produce more consistent and logical answers than generative readers.

All of these studies focus on improving machine comprehension reader performance using different approaches. However, more work is still needed in this area, and better models are expected to be developed in the future. Transformer models have become very popular in the field of machine-understanding readers in recent years. These models give successful results in solving many problems, especially in the field of natural language processing. Below are some machine-understanding reader exercises using transformer models:

BERT (Bidirectional Encoder Representations from Transformers) [16]: This work is an important example of the use of transformer models in the field of machine understanding readers. BERT is a set of models that are trained on an unformatted text corpus and then fine-tuned for different tasks. BERT has achieved high accuracy rates on datasets such as SQuAD.

RoBERTa (Robustly Optimized BERT Pretraining Approach) [17]: RoBERTa is a model that extends BERT's core pretraining methodology. RoBERTa is trained longer on a larger text corpus and then fine-tuned on datasets such as SQuAD. This model outperformed BERT.

ALBERT (A Lite BERT for Self-supervised Learning of Language Representations) [18]: ALBERT is a lightened version of BERT. ALBERT uses a number of innovative methods that keep BERT's encodings smaller while maintaining its performance. This model outperformed BERT on datasets such as SQuAD.

ELECTRA (Efficiently Learning an Encoder that Classifies Token Replacements Accurately) [19]: This model offers a more efficient learning method by

changing the learning process of BERT. ELECTRA misleadingly changes some words in the input text and then learns that the model must detect these misleading changes. This model achieved high accuracy rates on datasets such as SQuAD.

These studies show that transformer models can be used successfully in the field of machine understanding readers. It is envisaged that these models will be used in the future to further improve machine comprehension reader performance.

The T5 (Text-to-Text Transfer Transformer) [20] transformer model has made a significant contribution to the field of machine understanding readers. T5 is a model developed by Google that provides better performance by combining various natural language processing tasks into a single model.

In the machine understanding reader area, the T5 model is pre-trained for several different tasks. For example, the T5 model was trained on the SQuAD and Natural Questions (NQ) datasets and was among the best-performing models in both datasets.

The T5 is also a large model that enables fast and efficient learning. Therefore, large transformer models such as the T5 model may become even more common in the machine understanding reader field in the future and can be used to provide better performance. The BART (Bidirectional and Auto-Regressive Transformer) [21] transformer model has made a significant contribution to the field of machine understanding readers. BART is a model developed by Facebook and is a machine-understanding reader model that is both pre-trained and task-specific.

BART is a large model that successfully performs a variety of natural language processing tasks. In the machine understanding reader field, the BART model has achieved good results in experiments on the SQuAD, Natural Questions (NQ), and TriviaQA datasets.

The BART model is especially useful when there are data constraints. As a pre-trained language model, the model performs well on many datasets and achieves good results even when trained with a limited dataset.

The contribution of the BART model is especially important for researchers and application developers who want to achieve better results with less data in the machine-understanding reader area.

Federated Learning for Mobile Keyboard Prediction [9], In this study, Google used federated learning in mobile keyboard prediction to prevent learning data from being collected on a central server. In this way, the confidentiality of the data collected from users was preserved.

Federated Learning with Non-IID Data [10], In this study, Google demonstrated how the federated learning model can deal with different distributions of data (non-IID). This study showed that differences in data distribution were not a barrier during the data collection phase of federative learning.

Towards Federated Learning at Scale [11], In this study, Google presented a set of recommendations on systems design to enable large-scale federated learning applications.

In this study, we focus on an hybrid question answering system that can work on different sub systems. There exists examples of such hybrid systems in the literature for web services domain [23,24]. We observe studies in web services based systems with variety of focus area such as [25–30,32,38,39,41]. In this study, we utilize restful based web services for question and answering systems. There exits studies for data representation in different domains such as web usage mining [33,36,37]. In this study, we focus on the hybrid and federated question and answering systems. We leave the representation of the questions and answers as out-of-focus. We also observe studies that focus on quality of software systems [34,35].In this study, we mainly focus on building the question and answering system and leave out the analysis of software quality aspect for future work. There are studies focusing on misinformation and privacy policy violation in social networks [22,40]. Different from these studies, we focus on providing the right answer to the given questions. And we focus on transformer models trained based on the data available on the Web domain.

## 4 Proposed Methodology

In this study, we propose a hybrid questions and answering system as it is illustrated in Fig. 1. The proposed architecture enables users to utilize multiple question and answer systems in a transparent way via a single user interface. The proposed architecture also has a federation capability. The architecture includes modules that are responsible for merging the results coming from multiple subquestion and answer systems.

The proposed module has various modules: Distributor Module, Data Post Processing Module, Federator Module. We explain the details of each module below.

Distributor Module: This module is responsible for relaying incoming requests to corresponding question and answer subsystems. To do this, this module utilizes a mapping rule file. The mapping rule file maps the path for the subsystems with their description. It also includes the class names and the function names for the prediction functions of corresponding sub question answering systems. The distributed module also utilizes a properties file, which includes the configuration details of the proposed hybrid question and answering system. The distributed module is not dependent on the technology used in the subsystems. It is one layer above the subsystems. Hence, if the subsystems' implementations change, this will not affect the functioning of the Distributed Module. To archive this property, we utilize Facade Design pattern.

Data-Post Processing Module: The proposed framework will not only enable the hybrid prediction functions but also federated prediction functions. In case, the federation based prediction capability is requested by the end-user, then the post-processing module will process the results of each question and answering subsystem for federation capability. Here, the results will be processed for tasks like stop word removal, stemming etc.

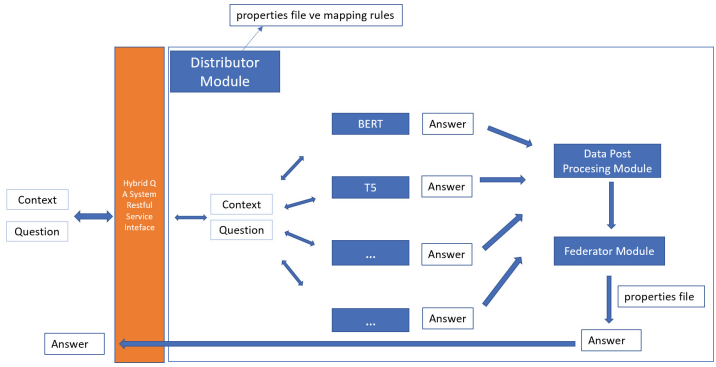


**Fig. 1.** Our approach on hybrid question answering architecture

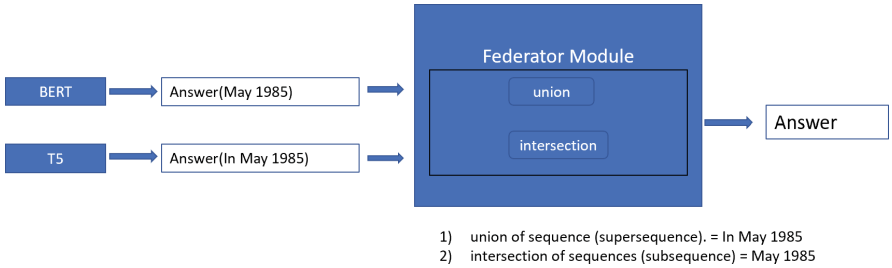


**Fig. 2.** Federator Module

Federator Module: This module is responsible federating the results coming from more than one question and answer systems. This module is illustrated in Fig. 2. It includes two sub modules, each is responsible for a federation method. Here, we only implemented union and intersection based result federation. We discuss the details of these methods below.

We denote a sequence s by $< a_1, a_2 \ldots, a_r >$, where $a_i$ is an wordset, which is also called an element of s. We denote an element (or an itemset) of a sequence by $x_1, x_2, \ldots, x_k$, where $x_i$ is an item. A sequence of length k is called a k-sequence.

If we depict the question-answering system as a sequence, we can show it as s sequence <hello world, explore the world>.

$a_1$= hello world, $a_2$= explore the world

$a_1$= {hello, world} $a_2$= {explore, the, world}

Union: The union of two sequences contains all elements in either sequence, without duplicates. For example, if sequence A = [hello, world] and sequence B = [explore, the, world], then their union A ∪ B = [hello, explore, the, world].

Intersection: The intersection of two sequences contains only the elements that appear in both sequences. For example, if sequence A = [hello, world] and sequence B = [explore, the, world], then their intersection A ∩ B = [world].

## 5 Prototype Application and Experimental Study

**Prototype:** We implemented the proposed software architecture using Pytorch library (version 1.13.1) by using Python programming language. We used T5 and BERT models as transformer model, T5tokenizer and BertTokenizer as tokenizer.

**Dataset:** In this study, a hybrid question answering systems study was performed using BERT and T5 transformer models. The TQUAD dataset was used in this study. TQUAD (Turkish Question Answering Dataset) is a question-answer dataset created for use in the field of Turkish natural language processing. This dataset has been prepared by researchers at Bilkent University for use in the field of Turkish natural language processing. TQUAD contains a total of 27,503 question-answer pairs, and these pairs were automatically extracted from the Turkish Wikipedia pages. This dataset can contribute to the advancement of research in Turkish natural language processing.

**Experimental Study and Discussion on the Results:** This study, by designing a hybrid question-answering system, We observed measure the execute time of the hybrid system by taking the results from many transformer models and hybrid system. The results from Transformer models, T5 and Bert, are obtained from sequence operations such as union and intersect, and the results according to execute time and standard deviation are shared in Table 1.

By sending a request thirty times in a row, we calculate the execute time and standard deviation of each response we obtained from Bert, t5 and the hybrid response, and show them in Table 1. Table 1 shows that the hybrid system offers negligible runtime by comparing the execution times by giving an additional load to the system. Our goal here is to integrate many transformer models into the hybrid system and to establish a more effective, useful system without losing time and we have examined the success of the hybrid system. The results indicate that the hybrid question and answering system has negligible processing overheads. Hence, it provides the hybrid question and answer system capability by providing a competitive execution performance.

**Table 1.** Execution Performance for Hybrid Question and Answering

| | Hybrid T5 (ms) | T5 (ms) | Hybrid BERT (ms) | BERT (ms) |
|---|---|---|---|---|
| avg | 699.58 | 692.64 | 92.19 | 86.71 |
| std | 24.45 | 17.84 | 7.89 | 5.44 |

## 6 Conclusion and Future Works

The study presented in this paper aims to address the challenges associated with using different question answering (QA) systems developed by various institutions and run on different platforms. The proposed software architecture enables QA systems to be used in a hybrid manner, allowing systems trained on different data sets and run on different platforms to be accessed through a single interface. The results of the study demonstrate the feasibility and usefulness of the proposed hybrid QA system framework software architecture, which can help improve the efficiency and accuracy of finding information from large amounts of text data. In addition, the study compares the effectiveness of T5 and BERT models in hybrid question answering systems. The results of the experimental study on the prototype application indicates that the proposed architecture has negligible processing overheads while providing the hybrid functionality.

Overall, the study highlights the importance of hybrid question answering systems and the potential of hybrid system to improve the accuracy of natural language processing models. The proposed framework software architecture can help address the current challenges associated with running QA systems on different platforms and using various technologies.

In the future, we plan on conducting experimental studies on federation capability of the proposed system architecture. We also plan on conducting further experimental studies to investigate the system performance as the size of the simultaneous users, using the system for question answer capabilities, increase.

## References

1. McMahan, B., Ramage, D.: Google Ai Blog. Communication-Efficient Learning with Federated Learning: An Overview (2020). https://ai.googleblog.com/2020/05/communication-efficient-learning-with.html
2. Tensorflow Federated: Machine Learning on Decentralized Data (2020). https://www.tensorflow.org/federated/. Accessed 16 Apr 2023
3. Federated AI Ecosystem-Collaborative Learning and Knowledge Transfer With Data Protection (2020). https://www.fedai.org/. Accessed 16 Apr 2023
4. PySyft: A Library for Encrypted, Privacy Preserving Machine Learning (2020). https://github.com/OpenMined/PySyft. Accessed 16 Apr 2023
5. PaddleFL: Federated Deep Learning in PaddlePaddle (2020). https://github.com/PaddlePaddle/PaddleFL. Accessed 16 Apr 2023
6. Nvidia Developer Blog: Federated Learning Powered by Nvidia Clara (2020). https://developer.nvidia.com/blog/federated-learning-clara/. Accessed 16 Apr 2023
7. McMahan, B., Moore, E., Ramage, D., Hampson, S., Arcas, B.: Communication efficient learning of deep networks from decentralized data. In: Proceedings of the 20th International Conference on Artificial Intelligence and Statistics (AISTATS) (2017)
8. Ekmefjord, M., et al.: Scalable federated machine learning with FEDN. In: 22nd IEEE International Symposium on Cluster, Cloud and Internet Computing (CCGrid) (2022). https://doi.org/10.1109/CCGrid54584.2022.00065

9. Hard, A., et al.: Federated learning for mobile keyboard prediction (2019). ArXiv:1811.03604
10. Zhao, Y., Li, M., Lai, L., Suda, N., Civin, D., Chandra, V.: Federated learning with non-IID data. ArXiv:1806.00582 (2018)
11. Bonawitz, K., et al.: Towards federated learning at scale: System design. ArXiv: 1902.01046 (2019)
12. Bakir, D., Aktas, M.: A systematic literature review of question answering: research trends, datasets, methods and frameworks. In: The 22nd International Conference on Computational Science and Its Applications (ICCSA), Malaga, Spain, 4–07 July 2022, pp. 1–16 (2022)
13. Bahdanau, D., Cho, K., Bengio, Y.: Neural machine translation by jointly learning to align and translate. arXiv:1409.0473 (2014)
14. Chen, D., Bolton, J., Manning, C.D.: A thorough examination of the CNN/daily mail reading comprehension task. In: Proceedings of the 54th Annual Meeting of the Association for Computational Linguistics (Volume 1: Long Papers), In ACL, pp. 2358–2367 (2016)
15. Seo, M.J., et al.: Bidirectional attention flow for machine comprehension. ICLR. arxiv:1611.01603 (2017)
16. Devlin, J., Chang, M.W., Lee, K., Toutanova, K.: Bidirectional encoder representations from transformers. In: Proceedings of the 2019 Conference of the North American Chapter of the Association for Computational Linguistics: Human Language Technologies, Volume 1 (Long and Short Papers), pp. 4171–4186. Association for Computational Linguistics (2019)
17. Liu, Y., Liu, D., Li, D., Lv, Y.: Robustly Optimized BERT Pretraining Approach. arXiv preprint arXiv:1907.11692 (2019)
18. Qi, Y., Wang, W., Zhang, B., Dauphin, Y.: A Lite BERT for Self-supervised Learning of Language Representations. arXiv preprint arXiv:2004.10948 (2020)
19. Xu, K., Zhu, W., Yang, Z., Bai, X.: Efficiently Learning an Encoder that Classifies Token Replacements Accurately. arXiv preprint arXiv:2009.13258 (2020)
20. Raffel, C., et al.: Exploring the limits of transfer learning with a unified text-to-text transformer. arXiv preprint arXiv:1910.10683 (2019)
21. Ghazvininejad, M., Levy, O., Liu, Y., Zettlemoyer, L.: A Knowledge-Grounded Autoencoder for Commonsense Inference. In: Proceedings of the 57th Conference of the Association for Computational Linguistics, pp. 5669–5674
22. Baeth, M.J., Aktas, M.S.: An approach to custom privacy policy violation detection problems using big social provenance data. Concurr. Comput. Pract. Exp. **30**(21) (2018)
23. Aktas, M.S., Fox, G.C., Pierce, M., Oh, S.: XML metadata services. Concurr. Comput. Pract. Exp. **20**(7), 801–823 (2008)
24. Aktas, M.S., Pierce, M.: High-performance hybrid information service architecture. Concurr. Comput. Pract. Exp. **22**(15), 2095–2123 (2010)
25. Fox, G.C., et al.: Real time streaming data grid applications. In: Davoli, F., Palazzo, S., Zappatore, S. (eds.) Distributed Cooperative Laboratories: Networking, Instrumentation, and Measurements. Signals and Communication Technology, pp. 253–267. Springer, Boston, MA (2006). https://doi.org/10.1007/0-387-30394-4_17
26. Tufek, A., Gurbuz, A., Ekuklu, O.F., Aktas, M.S.: provenance collection platform for the weather research and forecasting model. In: 2018 14th International Conference on Semantics, Knowledge and Grids (2018)
27. Aktas, M.S., Fox, G.C., Pierce, M.: Fault tolerant high performance information services for dynamic collections of grid and web services. Futur. Gener. Comput. Syst. **23**(3), 317–337 (2007)

28. Aydin, G., et al.: Building and applying geographical information system Grids. Concurr. Comput. Pract. Exp. **20**(14), 1653–1695 (2008)
29. Fox, G.C., et al.: Algorithms and the Grid. Comput. Vis. Sci. **12**, 115–124 (2009)
30. Aktas, M., et al.: ISERVO: implementing the international solid earth research virtual observatory by integrating computational grid and geographical information Web Services. Comput. Earthq. Phys. Simul. Anal. Infrastruct. Part **II**, 2281–2296 (2007)
31. Aydin, G., Aktas, M.S., Fox, G.C., Gadgil, H., Pierce, M., Sayar, A.: SERVOGrid complexity computational environments CCE integrated performance analysis. In: The 6th IEEE/ACM International Workshop on Grid Computing (2005)
32. Pierce, M.E., et al.: The QuakeSim project: Web services for managing geophysical data and applications. Earthq. Simul. Sourc. Tsunamis. 635–651 (2008)
33. Uygun, Y., Oguz, R.F., Olmezogullari, E., Aktas, M.S.: On the large-scale graph data processing for user interface testing in big data science projects. In: 2020 IEEE International Conference on Big Data (Big Data), pp. 2049–2056 (2020)
34. Sahinoglu, M., Incki, K., Aktas, M.S.: Mobile application verification: a systematic mapping study. In: Computational Science and Its Applications-ICCSA: 15th International Conference, Banff, AB, Canada, June 22–25, 2015, Proceedings. Part V, vol. 15(2015)
35. Kapdan, M., Aktas, M., Yigit, M.: On the structural code clone detection problem: a survey and software metric based approach. In: Computational Science and Its Applications-ICCSA,: 14th International Conference, Guimarães, Portugal, June 30-July 3, 2014, Proceedings. Part V, vol. 14 (2014)
36. Olmezogullari, E., Aktas, M.S.: Pattern2Vec: representation of clickstream data sequences for learning user navigational behavior. Concurr. Comput. Pract. Exp. **34**(9) (2022)
37. Olmezogullari, E., Aktas, M.S.: Representation of click-stream datasequences for learning user navigational behavior by using embeddings. In: 2020 IEEE International Conference on Big Data (Big Data), pp. 3173–3179 (2020)
38. Nacar, M.A.et al.: VLab: collaborative Grid services and portals to support computational material science. Concurr. Comput. Pract. Exp. **19**(12), 1717–1728 (2007)
39. Dundar, B., Astekin, M., Aktas, M.S.: A big data processing framework for self-healing internet of things applications. In: 2021 IEEE International Conference on Big Data (Big Data), pp. 2353–2361 (2021)
40. Baeth, M.J., Aktas, M.S.: Detecting misinformation in social networks using provenance data. In: 2017 13th International Conference on Semantics, Knowledge and Grids (2017)
41. Aktas, M., et al.: Implementing geographical information system grid services to support computational geophysics in a service-oriented environment. NASA Earth-Sun System Technology Conference, University of Maryland, Adelphi, Maryland (2005)