Proceedings of the OSACT 2022 Workshop @LREC2022, pages 130–135
Marseille, 20 June 2022

QQATeam at Qurán QA 2022: Fine-Tunning Arabic QA Models for Qurán QA Task

Basem H. Ahmed, Motaz K. Saad, Eshrag A. Refaee
Alaqsa University, The Islamic University of Gaza, Jazan University
Dept. of computer science, Faculty of Information Technology, Dept. of Info Technology & Security
basem@alaqsa.edu.ps, msaad@iugaza.edu.ps, erefaie@jazanu.edu.sa

Abstract

The problem of auto-extraction of reliable answers from a reference text like a constitution or holy book is a real challenge for the natural languages research community. Qurán is the holy book of Islam and the primary source of legislation for millions of Muslims around the world, which can trigger the curiosity of non-Muslims to find answers about various topics from the Qurán. Previous work on Question Answering (Q&A) from Qurán is scarce and lacks the benchmark of previously developed systems on a testbed to allow meaningful comparison and identify developments and challenges. This work presents an empirical investigation of our participation in the Qurán QA shared task (2022) that utilizes a benchmark dataset of 1,093 tuples of question-Qurán passage pairs. The dataset comprises Qurán verses, questions and several ranked possible answers. This paper describes the approach we follow with our participation in the shared task and summarises our main findings. Our system attained the best score at 0.63 pRR and 0.59 F1 on the development set and 0.56 pRR and 0.51 F1 on the test set. The best results of the Exact Match (EM) score at 0.34 indicate the difficulty of the task and the need for more future work to tackle this challenging task.

Keywords: Question Answering, Classic Arabic, Qurán question answering, fine-tuning, pre-trained models

1. Introduction

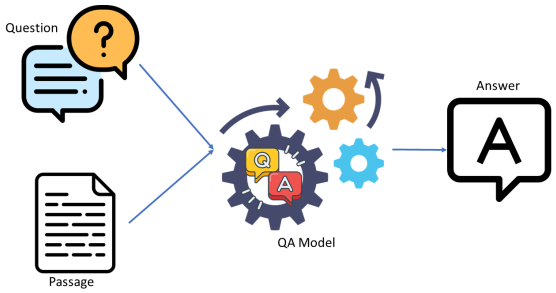
The enormous amount of data readily available and the advances in machine learning and computer-based systems in the past two decades have created the need for auto-extraction of answers for any given question. The domain of automatic extraction of answers to questions spanning various topics is a sub-field of Natural Language Processing (NLP). Specifically, QA is an NLP task concerned with querying information from content available in multiple formats, including structured and unstructured data (Bouziane et al., 2015). Research on QA is motivated by satisfying the users' need for obtaining answers across a variety of topics using computer-based and machine learning means to enhance the efficiency of this task.

A question-answering (QA) system is an application program that takes a user ś natural-language input question and attempts to return a precise answer (Prager, 2014). Figure 1 shows a typical QA model, where the inputs are the question and the passage, and the output is the answer extracted from the text.

QA as a task consists of three distinct NLP and IR subtasks: question analysis, passage retrieval, and answer extraction. Parsing the question is essential to extract its category, the type of answer required, and whether it is a name, place, quantity, or date. This early categorization of the question facilitates the answer extraction phase to select the best possible answer. The process can involve pre-processing steps like eliminating stop words, extracting named entities, and categorizing questions.

The application of QA to different languages has shown other performance mainly depending on how well-resourced is the language under consideration. According to the United Nations Educational, Scientific and Cultural Organisation (UNESCO), the Arabic language is the native tongue of more than 400 million people around the world.[1] The primary challenge when working on most NLP tasks for Arabic is the fact that Arabic is a morphological-rich language (Refaee, 2016). Another challenge is that, as compared to English, Arabic is still an under-resourced language with an increasing effort to address this issue.

Figure 1: Typical QA Model



QA can be applied to a wide range of text content, including web documents, constitutions and holy books. When it comes to Qurán as the major source of legislation for nearly 1.9 billion Muslims (Meters, 2022) and the way it attracts the curiosity of non-Muslims, QA becomes even more interesting. Several attempts have been made in this area to tackle the difficulty of auto-

[1] Available at https://en.unesco.org/commemorations/worldarabiclanguageday Accessed on 09/04/2022.

matically extracting answers regarding different questions from Qurán verses. However, previous QA work on Qurán is limited and lacks a reasonable and meaningful benchmark on a common testset. For this purpose, a shared task, namely Quraán QA 2022, has been launched.[2] The added value of the shared task, besides allowing several parallel attempts to develop QA systems, is the availability of a benchmark testset that has been released as a part of the shared task (Malhas et al., 2022). The competition of several teams worldwide on a benchmark test set allows for achieving remarkable improvements and identifying challenges associated with QA on Qurán. The work (Malhas et al., 2022) presents an overview of the shared task and outlines the approaches employed and results attained by the participating teams.

2. Related Work

The task of auto-extraction of answers for a given question is not new for Natural Language Processing (NLP) researchers. Literature has revealed considerable interest in this task for more than two decades. Research in this area has been highly motivated by computer-based methods and tools to extract reliable answers from a given text automatically. In most cases, the text used to extract answers is considered a reference, like the constitution or the holy textbook. However, the source for extracting answers can be merely news websites in other cases. The results of question answering have shown to be varied across different languages. This section outlines the previous work that has addressed the problem of automatic question-answering in Arabic.

In early work by (Hammo et al., 2002), traditional Information Retrieval (IR) techniques were used with an NLP approach. Specifically, the authors used a keyword matching strategy and matching simple structures extracted from both the question and the candidate documents selected by the IR system. The authors utilized a morphological analyzer and an existing tagger to identify proper names and build lexical entries for them. As for word-level, they used root stemming and identified the query type (What, when, where, who, etc.).

(Magdy and Shaheen, 2012) presented a survey on exciting efforts to tackle the main challenges associated with the question answering task in Arabic. The work outlined the approaches and tools utilized, including the early classification of the questions into Name, Date, and Quantity to determine the question type. This classification involves defining the type of a given question according to the question word used to extract the expected answer type, question focus and important question keywords. Common pre-processing steps involved question tokenization, normalization, removing stop words and stemming. A common practice was to determine the question focus, which is the proper noun phrase that the question mainly revolves around, which usually leads to choosing the answer type based on question words like who and when. Another common practice revealed by the authors is the utilization of Named Entity (NE) recognition tools. In addition, the passage retrieval technique utilized in QA mainly was the co-occurrence of question and answer keywords within the same context. Finally, semantic reasoning was accomplished by exploiting existing platforms like Amine to score and rerank the retrieved passages semantically using concept graphs to find the most relevant answer passage. Specifically, they used the semantic similarity between the focus of the question and the candidate's answer using N-grams.

Arabic has limited resources for the QA task, unlike English and other well-resourced languages where multiple large QA datasets are freely available (Rajpurkar et al., 2016). The linguistic resources are even more scarce when finding a dataset of Qurán versus annotated for QA. A recent effort (Malhas and Elsayed, 2020) has addressed this issue and introduced a publicly available reusable test collection for Arabic question answering on the Holy Qurán, namely AyaTEC. According to the authors, their test collection for verse-based question answering on the Holy Qur'an serves as a standard experimental testbed for QA. The dataset AyaTEC includes 207 questions with their corresponding 1,762 answers, spanning 11 topic categories. The authors stated that the dataset of the Holy Qurán targets the information needs of both curious and sceptical users. They proposed several evaluation measures to support the different types of questions and the nature of verse-based answers while integrating the concept of partial matching of answers in the evaluation. The dataset was used in the shared task to allow multiple systems to be implemented and compared.

In (Abdelnasser et al., 2014), the authors proposed an Arabic QA system specializing in the Holy Qurán. The system takes an Arabic question about the Qurán, retrieves the most relevant Qurán verses, and then extracts the passage that contains the answer from the Qurń. They utilized the Quranic Corpus Ontology to obtain and manually revise 1200 data instances. The authors reported up to 85% accuracy using the top-3 results.

(Hamdelsayed and Atwell, 2016) presented a rule-based system for the Holy Qurán that retrieves the correct verse from the Holy Qurán. The authors utilized their dataset and reported an improvement due to simple pre-processing of removing stop words.

In (Hamed and Ab Aziz, 2016), the authors used an Existing English translation of Qurán verses to develop a system utilizing neural networks (NN) for QA. They expanded the question by using WordNet. In addition, they utilized the NN classifier to reduce the retrieval of irrelevant verses using the word N-gram technique. The following step included ranking the re-

[2]Available at `shorturl.at/dlFH6` Accessed on 12/03/2022.

trieved verses based on the highest similarity score to fulfil the user question. The authors reported an F-score up to 87% on classification and recommended employing classification as an initial stage for retrieving verses as answers for a given question.

the authors in (Mozannar et al., 2019a) tackled the problem of open domain factual Arabic question answering (QA) using Wikipedia as our knowledge source. The authors reported that Open-domain QA for Arabic entails three challenges: annotated QA datasets in Arabic, large scale efficient information retrieval and machine reading comprehension. They addressed the first challenge by compiling The Arabic Reading Comprehension Dataset (ARCD). To address the second and the third challenge, the authors presented an open domain question-answering in Arabic (SOQAL) that is based on two components: (1) a document retriever using a hierarchical TF-IDF approach and (2) a neural reading comprehension model using the pre-trained bi-directional transformer BERT.

(Su et al., 2019) addressed the problem of generalizing QA models with pre-trained models fine-tuning. They fine-tuned a large pre-trained language model (XLNet) on multiple RC datasets. The results suggest that fine-tuning is effective.

On the other side, several attempts have been made to address the problem of QA in Arabic in general, not only in Qurán. Recent work (Alsubhi et al., 2021) has thoroughly highlighted the task of QA in Arabic. The authors evaluated the state-of-the-art pre-trained transformers models for Arabic QA using four datasets (Arabic-SQuAD, ARCD, AQAD, and TyDiQA-GoldP). They fine-tuned three pre-trained models (AraBERTv2-base, AraBERTv0.2-large, and AraELECTRA). The authors address the impact of the size and quality of the dataset on the performance of their proposed QA model. They also tried to improve the performance by fine-tuning hyper-parameters. The authors reported that the best F-score was 61%, obtained using AraBERTv0.2-large on Arabic-SQuAD dataset.

A more comprehensive view of QA in Arabic can be found in a recent survey (Alwaneen et al., 2021). To sum up, previous works lack benchmark comparison on a standard testbed.

3. Dataset

The shared-task data comprises 1,093 tuples of question-passage pairs coupled with their extracted answers to constitute 1,337 question-passage-answer triplets (Malhas and Elsayed, 2022). The benchmark dataset has been accessible for the teams registered in the competition (Malhas and Elsayed, 2020). The dataset distribution into training, development and test sets is shown below.

Dataset	%	# Question Passage Pairs	# Question Passage Answer Triplets
Training	65%	710	861
Dev	10%	109	128
Test	25%	274	348
All	100%	1,093	1,337

4. Approach

In this task, the approach uses Arabic QA pre-trained models and fine-tunes them with the Qurán QA dataset.

We use two pre-trained Arabic QA models listed in Table1, hosted on Hugging Face (Wolf et al., 2020). We used these models because they are the only existing pre-trained models that support the Arabic language on Hugging Face. In addition, these models can be fine-tuned easily.

Table 1: Arabic QA pre-trained Models

Arabic QA Model	Trained on	Reference
Salti Ara-Electra base fine-tuned ARCD (AraElectra-ARCD)	Arabic Reading Comprehension Dataset (ARCD) composed of 1,395 questions posed by crowd-workers on Wikipedia articles	(Mozannar et al., 2019b)
Wissam Antoun Ara-Electra base Artydiqa (AraElectra-Artydiqa)	TyDi QA is a question answering dataset covering 11 topologically diverse languages with 204K question-answer pair (Clark et al., 2020)	(Antoun et al., 2020)

The fine-tuning is done on the training data and the training, development, and augmented data merge. We manually applied data augmentation to the training and development parts of the dataset by paraphrasing only the question part on the QA dataset. Paraphrasing is done by changing word order, using different synonyms when asking about an object, using function words, and using a different questioning tool. Our hypothesis here is that data augmentation may help fine-tune the model to correct answers to different question forms in the test set. Augmented Data is described in Table 2, and can be found on `https://github.com/motazsaad/Quran-QA`.

Besides fine-tuning the two pre-trained models, we combine these two models and choose the best scores from both models for the answers obtained from them. So we made three submissions to the shared task. The first one uses *AraElectra-ARCD*, and the second uses

Table 2: Augmented Data

Dataset	Size
Training Question Passage Answer Triplets (training only)	861
Training Question Passage Answer Triplets (training and Dev)	989
augmented Question Passage Answer Triplets (training and Dev)	657
Training Question Passage Answer Triplets (training and dev and augmented)	1646

AraElectra-Artydiqa. The third attempt uses the Hybrid model, in which the two fined-tuned models are used to get the answers with their scores (weights), and then answer scores from both models are normalized and ranked together. The top 5 answers that have the highest scores are selected.

We use Colab Pro for fine-tuning, and the "*Salti Ara-Electra base fine-tuned ARCD*" model stopped at epoch 8, while the "*Wissam Antoun Ara-Electra base Artydiq*" model stopped at epoch 4.

Data is pre-processed by applying the normalization function that is provided by the maintainer of this shared task `https://gitlab.com/bigirqu/quranqa/-/blob/main/code/quranqa22_eval.py`, where the stopwords (only Arabic prepositions), and punctuation are removed. In addition, a predefined list of prefixes is removed.

5. Results and Discussion

Table 3delivers the performance of the QA pre-trained models fine-tuned on train data and tested on Dev data. Table 4 shows the performance of the QA pre-trained models fine-tuned on train, Dev and augmented data and tested on Test data. Figures 2 and 3 show the performance of *"AraElectra-ARCD"* and *"AraElectra-Artydiqa"* models, respectively. The figures indicate the pRR, Exact Match and F1@1 metrics with training epochs.

Table 3: Dev data Results using fine-tuned using training data

Model	pRR	Exact Match	F1@1
AraElectra-ARCD	0.60544	0.33027	0.57807
AraElectra-Artydiqa	0.61828	0.32110	0.57804
Hybrid	**0.62571**	**0.33944**	**0.59145**

Table 4 shows the test data results fine-tuned using Dev and augmented data. It can be noted from the Table 3 that the best fine-tuned QA model is the combined model with the following scores 0.62 pRR, 0.33 Exact Match, 0.59 F1@1. On the other hand, Table 4 shows that the best fine-tuned QA model is *AraElectra-Artydiqa*, with the following scores; 0.55 pRR, 0.24 Exact match, and 0.51 F1@1. The result suggests that

Figure 2: Performance of AraElectra-ARCD model

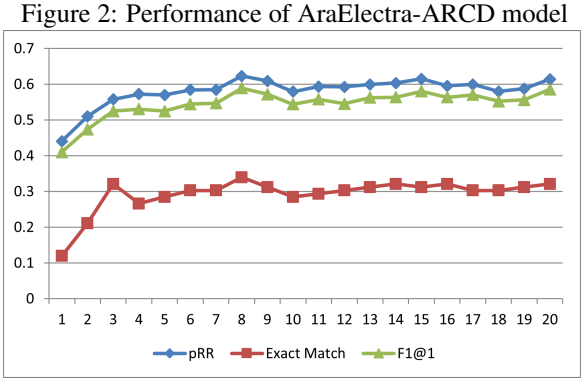
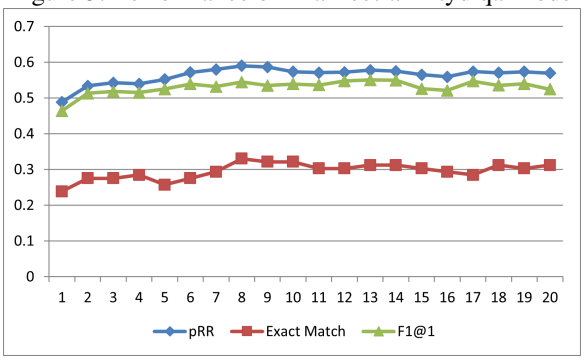


Figure 3: Performance of AraElectra-Artydiqa Model



the combined model worked well when applied to the Dev dataset and obtained the best results, but the result was not the same on the model used on the test set. The best performing model on the test set was the *AraElectra-Artydiqa*, trained initially on more extensive data than the *AraElectra-ARCD* model. The performance of *AraElectra-Artydiqa* was even better than the combined model on the test set. This suggests that picking the trained model on a large dataset can be best for fine-tuning.

Comparing the best results in the two tables, we can see that When the models are applied to the test set (Table 4), the pRR score dropped 7%, the exact match dropped 9%, and the F1@1 dropped 8%. This performance is expected and suggests that the models need more fine-tuning, and the training data should be enlarged. Moreover, the domain of both models is Wikipedia, which is away from the Qurán QA domain, and the fine-tuning was not enough to get promising results because the QA training data is small.

6. Conclusion and Future Work

The ability to automatically extract answers from a user input natural text is one of the leading NLP tasks with plenty of real-life applications. QA task comes with several challenges, and performance on this task can vary across different languages. Obtaining answers from references like a constitution or holy textbooks can be more challenging than getting answers from other sources like news websites. That includes the limited linguistic resources available (i.e., annotated

Table 4: Test data Results fine-tuned using training and Dev and augmented data

Model	pRR	Exact Match	F1@1
AraElectra-ARCD	0.52600	0.22269	0.46228
AraElectra-Artydiqa	**0.55889**	**0.24370**	**0.51326**
Hybrid	0.53486	0.23109	0.49997

data) and the need to have the answers as accurate as possible. For example, Qurán is the primary source of legislation in Islam and having systems that accurately extract answers regarding laws and Islamic-based concepts is critical.

This paper presents the participation of our team, namely QQATeam, in the Qurán shared task (2022). The shared task released a benchmark Qurán dataset of 1,093 tuples of question-passage pairs. The shared task aims to allow different teams to participate in developing other systems using the benchmark dataset in combination with various NLP resources, tools, and techniques that the teams wish to employ. Unlike previous work that has been done on QA from Qurán lacks the meaningful comparison of different QA approaches on a shared testset to allow identifying a baseline performance. The work produced by the shared task can help identify the QA task's state-of-the-art performance and reveal the opportunities and challenges associated with this task. By fine-tuning pre-trained models, our system attained the best performance at 0.56 pRR and 0.51 F1. A detailed explanation of approaches used and results achieved by participating teams can be found at (Malhas et al., 2022).

The overall results indicate the need for further developments to tackle the challenges identified in the QA task on the Qurán text. Future directions can involve using Information Retrieval (IR) to improve the results by passing the question as a query and ranking passages according to this query. Then the question and the top-ranked passage can be fed to the QA model. In addition, Qurán commentaries for each verse to determine the best verse that contains the answer.

7. Bibliographical References

Abdelnasser, H., Ragab, M., Mohamed, R., Mohamed, A., Farouk, B., El-Makky, N. M., and Torki, M. (2014). Al-bayan: an arabic question answering system for the holy quran. In *Proceedings of the EMNLP 2014 Workshop on Arabic Natural Language Processing (ANLP)*, pages 57–64.

Alsubhi, K., Jamal, A., and Alhothali, A. (2021). Pre-trained transformer-based approach for Arabic question answering: A comparative study. *arXiv preprint arXiv:2111.05671*.

Alwaneen, T. H., Azmi, A. M., Aboalsamh, H. A., Cambria, E., and Hussain, A. (2021). Arabic question answering system: a survey. *Artificial Intelligence Review*, pages 1–47.

Antoun, W., Baly, F., and Hajj, H. (2020). Araelectra: Pre-training text discriminators for arabic language understanding.

Bouziane, A., Bouchiha, D., Doumi, N., and Malki, M. (2015). Question answering systems: Survey and trends. *Procedia Computer Science*, 73:366–375. International Conference on Advanced Wireless Information and Communication Technologies (AWICT 2015).

Clark, J. H., Choi, E., Collins, M., Garrette, D., Kwiatkowski, T., Nikolaev, V., and Palomaki, J. (2020). TyDi QA: A benchmark for information-seeking question answering in typologically diverse languages. *Transactions of the Association for Computational Linguistics*.

Hamdelsayed, M. A. and Atwell, E. (2016). Islamic applications of automatic question-answering.

Hamed, S. K. and Ab Aziz, M. J. (2016). A question answering system on holy quran translation based on question expansion technique and neural network classification. *J. Comput. Sci.*, 12(3):169–177.

Hammo, B., Abu-Salem, H., Lytinen, S. L., and Evens, M. (2002). QARAB: A: Question answering system to support the arabic language. In *Proceedings of the ACL-02 workshop on Computational approaches to semitic languages*.

Magdy, A. and Shaheen, M. (2012). A survey of arabic question answering: Challenges, tasks, approaches, tools, and future trends. 10.

Malhas, R. and Elsayed, T. (2020). AyaTEC: building a reusable verse-based test collection for arabic question answering on the holy qur'an. *ACM Transactions on Asian and Low-Resource Language Information Processing (TALLIP)*, 19(6):1–21.

Malhas, R. and Elsayed, T. (2022). official repository of Qur'an QA Shared Task. `https://gitlab.com/bigirqu/quranqa`, February.

Malhas, R., Mansour, W., and Elsayed, T. (2022). Qurán QA 2022: Overview of the first shared task on question answering over the holy qurán. In *Proceedings of the 5th Workshop on Open-Source Arabic Corpora and Processing Tools (OSACT5) at the 13th Language Resources and Evaluation Conference (LREC 2022)*.

Meters, C. (2022). Religion of the world. `https://countrymeters.info/en/World#religion`, journal=Religion of the World, Apr.

Mozannar, H., Maamary, E., El Hajal, K., and Hajj, H. (2019a). Neural Arabic question answering. In *Proceedings of the Fourth Arabic Natural Language Processing Workshop*, pages 108–118, Florence, Italy, August. Association for Computational Linguistics.

Mozannar, H., Maamary, E., El Hajal, K., and Hajj, H. (2019b). Neural Arabic question answering.

In *Proceedings of the Fourth Arabic Natural Language Processing Workshop*, pages 108–118, Florence, Italy, August. Association for Computational Linguistics.

Prager, J. (2014). Question answering. https://www.oxfordhandbooks.com/view/10.1093/oxfordhb/9780199573691.001.0001/oxfordhb-9780199573691-e-003, journal=Oxford Handbooks Online, Apr.

Rajpurkar, P., Zhang, J., Lopyrev, K., and Liang, P. (2016). SQuAD: 100,000+ questions for machine comprehension of text. In *Proceedings of the 2016 Conference on Empirical Methods in Natural Language Processing*, pages 2383–2392, Austin, Texas, November. Association for Computational Linguistics.

Refaee, E. A. A. (2016). *Sentiment analysis for microblogging platforms in Arabic*. Ph.D. thesis, Heriot-Watt University.

Su, D., Xu, Y., Winata, G. I., Xu, P., Kim, H., Liu, Z., and Fung, P. (2019). Generalizing question answering system with pre-trained language model fine-tuning. In *Proceedings of the 2nd Workshop on Machine Reading for Question Answering*, pages 203–211, Hong Kong, China, November. Association for Computational Linguistics.

Wolf, T., Debut, L., Sanh, V., Chaumond, J., Delangue, C., Moi, A., Cistac, P., Rault, T., Louf, R., Funtowicz, M., Davison, J., Shleifer, S., von Platen, P., Ma, C., Jernite, Y., Plu, J., Xu, C., Scao, T. L., Gugger, S., Drame, M., Lhoest, Q., and Rush, A. M. (2020). Transformers: State-of-the-art natural language processing. In *Proceedings of the 2020 Conference on Empirical Methods in Natural Language Processing: System Demonstrations*, pages 38–45, Online, October. Association for Computational Linguistics.