Proceedings of the OSACT 2022 Workshop @LREC2022, pages 79–87
Marseille, 20 June 2022

Qur'an QA 2022: Overview of The First Shared Task on Question Answering over the Holy Qur'an

Rana Malhas, Watheq Mansour, Tamer Elsayed
Qatar University, Doha, Qatar
{rana.malhas, wm1900793, telsayed}@qu.edu.qa

Abstract

Motivated by the resurgence of the machine reading comprehension (MRC) research, we have organized the first Qur'an Question Answering shared task,"Qur'an QA 2022". The task in its first year aims to promote state-of-the-art research on Arabic QA in general and MRC in particular on the Holy Qur'an, which constitutes a rich and fertile source of knowledge for Muslim and non-Muslim inquisitors and knowledge-seekers. In this paper, we provide an overview of the shared task that succeeded in attracting 13 teams to participate in the final phase, with a total of 30 submitted runs. Moreover, we outline the main approaches adopted by the participating teams in the context of highlighting some of our perceptions and general trends that characterize the participating systems and their submitted runs.

1. Introduction

The Holy Qur'an is sacredly held by more than 1.9B Muslims across the world.[1] It is the major source of knowledge, teachings, wisdom, and legislation in Islam. This makes it a rich and fertile source for Muslim and non-Muslim knowledge-seekers pursuing answers to questions raised for learning, out of curiosity, or skepticism. The Qur'an is composed of 114 chapters (Suras) and 6236 verses (Ayas) of different lengths, with a total of about 80k Arabic words. The words, revealed more than 1,400 years ago, are in *classical Arabic* (CA) (Atwell et al., 2011). It is a phenomenal yet challenging document collection due to its long-chained anaphoric-structures and unstructured topic diversity. A Qur'anic verse may relate to one or more topics, and the same topic may be tackled in different verses and chapters, but in variant contexts.

Extractive question answering (QA) approaches are recently being formulated as machine reading comprehension (MRC) tasks (Chen et al., 2017; Chen, 2018). MRC was initially used (in the 1970s) to evaluate the task of language understanding by computer systems (Chen, 2018). Given a passage of text, a system is evaluated based on its ability to correctly answer a set of questions over the given text. The resurgence of the MRC field (after being dormant for decades) is mainly attributed to the release of relatively large MRC datasets (Rajpurkar et al., 2016; Joshi et al., 2017) that facilitated exploiting and training of intelligent deep learning neural MRC models with better understanding capability. We believe that such MRC intelligence should be harnessed to address the permanent interest in the Holy Qur'an and the information needs of its inquisitors and knowledge seekers (Atwell et al., 2011; Bashir et al., 2021).

To this end, the main goal of the Qur'an QA shared task is to promote state-of-the-art research on Arabic QA and MRC tasks over the Holy Qur'an. At the same time, the main objective is to foster a common experimental test-bed for systems to showcase and benchmark their performance (Malhas and Elsayed, 2020).

To encourage quality participation in the task, we allotted four awards. The first 3 awards were $500, $350, and $250 allotted for the top 3 ranked teams, respectively, given that their papers are accepted. The fourth one was $150 allotted for the best paper.

The Qur'an QA shared task in its first round (2022)[2] has succeeded in attracting thirty teams to sign up for the task. In the final phase, 13 teams participated, with a total of 30 submitted runs on the test set. Ten out of the thirteen teams submitted system description papers.

The paper is organized as follows. In Section 2, we present the task description and briefly discuss its main challenges. Then we present the dataset used in the shared task in Section 3. The evaluation setup and evaluation measures, in addition to the leaderboard developed on Codalab, are described in Section 4. The results are presented in Section 5, which we follow with an overview of the methods and an analysis of some trends and results in Section 6. We conclude with final thoughts in Section 7.

2. Task Description

Our task is defined as follows. Given a Qur'anic passage that consists of consecutive verses in a specific surah of the Holy Qur'an, and a free-text question posed in Modern Standard Arabic (MSA) over that passage, a system is required to extract an answer to that question. The answer must be a *span* of text extracted from the given passage. The question can be a factoid or non-factoid question. Examples are shown in Figure 1 and 2. As the shared task is introduced for the first time, the task this year was relatively simplified such that a system may find *any* correct answer

[1] https://en.wikipedia.org/wiki/Islam_by_country

[2] https://sites.google.com/view/quran-qa-2022

الفقرة القرآنية Qur'anic Passage (38:41-44)
وَٱذۡكُرۡ عَبۡدَنَآ أَيُّوبَ إِذۡ نَادَىٰ رَبَّهُۥٓ أَنِّي مَسَّنِيَ ٱلشَّيۡطَٰنُ بِنُصۡبٖ وَعَذَابٍ. ٱرۡكُضۡ بِرِجۡلِكَ هَٰذَا مُغۡتَسَلُۢ بَارِدٞ وَشَرَابٞ. وَوَهَبۡنَا لَهُۥٓ أَهۡلَهُۥ وَمِثۡلَهُم مَّعَهُمۡ رَحۡمَةٗ مِّنَّا وَذِكۡرَىٰ لِأُوْلِي ٱلۡأَلۡبَٰبِ. وَخُذۡ بِيَدِكَ ضِغۡثٗا فَٱضۡرِب بِّهِۦ وَلَا تَحۡنَثۡۗ إِنَّا وَجَدۡنَٰهُ صَابِرٗاۚ نِّعۡمَ ٱلۡعَبۡدُ إِنَّهُۥٓ أَوَّابٞ.
السؤال/ Question: من هو النبي المعروف بالصبر؟
الإجابة الذهبية / Gold Answer:
1. أَيُّوبَ

Figure 1: An example of a factoid question whose answer is a single span of text.

الفقرة القرآنية Qur'anic Passage (74:32-48)
كَلَّا وَٱلۡقَمَرِ. وَٱلَّيۡلِ إِذۡ أَدۡبَرَ. وَٱلصُّبۡحِ إِذَآ أَسۡفَرَ. إِنَّهَا لَإِحۡدَى ٱلۡكُبَرِ. نَذِيرٗا لِّلۡبَشَرِ. لِمَن شَآءَ مِنكُمۡ أَن يَتَقَدَّمَ أَوۡ يَتَأَخَّرَ. كُلُّ نَفۡسِۭ بِمَا كَسَبَتۡ رَهِينَةٌ. إِلَّآ أَصۡحَٰبَ ٱلۡيَمِينِ. فِي جَنَّٰتٖ يَتَسَآءَلُونَ. عَنِ ٱلۡمُجۡرِمِينَ. مَا سَلَكَكُمۡ فِي سَقَرَ. قَالُواْ لَمۡ نَكُ مِنَ ٱلۡمُصَلِّينَ. وَلَمۡ نَكُ نُطۡعِمُ ٱلۡمِسۡكِينَ. وَكُنَّا نَخُوضُ مَعَ ٱلۡخَآئِضِينَ. وَكُنَّا نُكَذِّبُ بِيَوۡمِ ٱلدِّينِ. حَتَّىٰٓ أَتَىٰنَا ٱلۡيَقِينُ. فَمَا تَنفَعُهُمۡ شَفَٰعَةُ ٱلشَّٰفِعِينَ.
السؤال/ Question: ما هي الدلائل التي تشير بأن الانسان مخير؟
الإجابات الذهبية / Gold Answers:
1. لِمَن شَآءَ مِنكُمۡ أَن يَتَقَدَّمَ أَوۡ يَتَأَخَّرَ
2. كُلُّ نَفۡسِۭ بِمَا كَسَبَتۡ رَهِينَةٌ

Figure 2: An example of a non-factoid question whose answers are two spans of text.

from the accompanying passage, even if the question has more than one answer in the given passage.

The systems of participating teams were required to return up to 5 potential answers, ranked from 1 (top/best) to 5 (lowest), from the accompanying passage for the given question. Therefore, the system is rewarded for returning any of the correct answers as higher as possible in the returned ranked list of answers.

The task is relatively challenging given the challenges of the Qur'an that are posed by its long anaphoric verse structures and unstructured topic diversity. Moreover, the vocabulary mismatch problem (that is typical for any QA or MRC task) is compounded here due to the literary style of the Qur'anic Classical Arabic. Such style would tend to make the answers evidence-based rather than natural language answers, especially for non-factoid questions. For example, answers to evidence questions (Figure 2) or yes/no questions necessitate finding evidence that asserts or negates the question. Such evidence answers are highly likely to include Classical Arabic words that have no overlap with the posed MSA question.

More importantly, the sacredness and unstructured topic diversity of the Qur'an pose a very critical challenge to machine learning (ML) and artificial intelligence (AI) approaches, not to generate results out of their intended context. As such, we should be extra cautious of using the results of learned models without the involvement of Qur'an scholars. (Bashir et al., 2021) discuss the caveats and potential pitfalls in Qur'anic NLP research that we should be wary of.

3. Dataset

The dataset for the task is the Qur'anic Reading Comprehension Dataset (or *QRCD* for short) (Malhas and Elsayed, 2022). It is an extension of the *AyaTEC* dataset (Malhas and Elsayed, 2020). It is composed of 1,093 tuples of question-passage pairs that are coupled with their extracted answers to constitute 1,337 question-passage-answer triplets. It is split into training (65%), development (10%), and test (25%) sets as shown in Table 1. *QRCD* is formatted as a JSON Lines (JSONL) file; each line is a JSON object that comprises a question-passage pair, along with its answers extracted from the accompanying passage.

Each Qur'anic passage in *QRCD* may have more than one occurrence; and each passage occurrence is paired with a different question. Likewise, each question in *QRCD* may have more than one occurrence; and each question occurrence is paired with a different Qur'anic passage. The source of the Qur'anic text in *QRCD* is the Tanzil project,[3] which provides verified versions of the Holy Qur'an in several scripting styles. Although we have chosen the simple-clean text style of Tanzil v1.0.2 for processing, it was later brought to our attention that the Uthmani orthography[4] should be used when quoting and printing Qur'an verses (Al-Azami, 2020).

[3] https://tanzil.net/download/

[4] Al-rasm al-Uthmani (or rasm al-mushaf) is the convention adopted for writing the Qur'anic text during the ruling of Caliph Uthman bin Affan (Al-Azami, 2020; Bashir et al., 2021).

Table 1: Distribution of question-passage pairs in QRCD

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

4. Evaluation Setup

4.1. Evaluation Measures

The task is viewed as a ranking task. Therefore, we used three rank-based evaluation measures, namely, partial Reciprocal Rank (pRR), Exact Match (EM), and $F_1@1$. We choose pRR as the main evaluation measure to give credit to a QA system that may retrieve an answer that is not necessarily at the first rank and/or *partially* (i.e., not exactly) match one of the gold answers (Malhas and Elsayed, 2020). However, EM and $F_1@1$ are applied only to the *top* predicted answer. While EM measure is binary, i.e., gives 1/0 score based on whether or not the top predicted answer *fully* matches the gold answer, $F_1@1$ measure is computed based on the overlap between the top predicted answer and the best matching gold answer (Rajpurkar et al., 2016). To get an overall evaluation score, each of the above measures is averaged over all questions.

To enable public research and allow participants to evaluate their runs, we made the dataset publicly available over the official repository of the shared task.[5] We also shared the following scripts with the teams:

- A reader script, which simply reads the tuples in the QRCD dataset.
- A submission checker script, which verifies the correctness of the run file. The run file should be in JSON format and has a list of passage-question ids along with their respective ranked lists of returned answers.
- An evaluation (scorer) script, which evaluates the run file according to the adopted different evaluation measures.

4.2. Leaderboard and Run Submission

For ease of submission and comparison, we hosted the task on Codalab platform.[6] A participating team is required to write answers to all questions (of the development or the test sets) in one file in a specific format, denoted as a "run file" or a "run" in short. A run typically constitutes the results of a different system or a model. We allow participants to submit up to 30 runs in the development phase, and up to 3 runs only in the test phase. Each team was allowed to submit its runs by its designated team leader only. In both development and test phases, we rank the teams based on their best run. However, teams were encouraged to describe their systems created for this task in their papers.

To give participants a reference point over the leaderboard, we created a simple (and *naïve*) baseline that just answers each question by returning the corresponding full passage as an answer.

5. Results

Thirty teams registered for the task. We received more than 100 submissions in both phases; 30 submissions were for the test phase. Among the 30 teams registered for the task, 13 teams participated in the final (test) phase. Table 2 presents the names of participating teams in the task, their size in terms of number of members, and their affiliations. We noticed a wide diversity in the participating teams. The participants are affiliated with 21 different institutes; all but one are universities. We also note that six of the teams are composed of at least two collaborating institutes.

The pRR score of the best run per team is used to rank the teams. Table 3 shows the evaluation results of the best run of each team ranked by pRR metric. Also, Table 4 illustrates the evaluation results of all submitted runs ranked by pRR metric as well. We underline the rows of the median runs. The top three teams are TF200, TCE, and QQATeam. The highest pRR score is 0.586 and the highest $F_1@1$ is 0.537, and both of them were achieved by TF200. However, the highest EM score is 0.269 which is achieved by TCE.

We noticed that all teams used transformer-based models that support Arabic to build their systems. The top teams used AraBERT and AraElectra in their systems, demonstrating high performance for these models. More details and analysis about the used approaches and their performance are in Section 6.

To see the performance distribution of all submitted runs across different test questions, Figure 3 shows the boxplots for them. It illustrates very diverse performance across the test questions for most of the runs.

6. Methods and Analysis

In this section, we give an overview of the main approaches adopted by the 13 participating teams in their 30 submitted runs on the test set. We do that in the context of highlighting some of our perceptions and general trends that characterize the participating systems and their submitted runs.

[5] https://gitlab.com/bigirqu/quranqa

[6] https://codalab.lisn.upsaclay.fr/competitions/2536

Table 2: Participating teams in Qur'an QA 2022

Team	Size	Affiliations
TF200	2	King Fahd University of Petroleum and Minerals
stars (Wasfey et al., 2022)	4	Tactful AI, Alexandria University, University of central Punja, Al-Azhar.
TCE (ElKomy and Sarhan, 2022)	3	Tanta University
QQATeam (Ahmed et al., 2022)	3	Alaqsa University, The Islamic University of Gaza, Jazan University
eRock (Aftab and Malik, 2022)	2	Punjab University
GOF (Mostafa and Mohamed, 2022)	3	Helwan University
LARSA22 (Mellah et al., 2022)	6	National School of Applied Sciences (ENSAH), Superior School of Technology (Meknes)
Rootroo	2	École Normale Supérieure, University of Helsinki
UM6P	3	Mohammed VI Polytechnic University
DTW (Premasiri et al., 2022)	3	University of Wolverhampton, Hamad Bin Khalifa University
SMASH (Keleg and Magdy, 2022)	2	The University of Edinburgh
LK2022 (Alsaleh et al., 2022)	6	University of Leeds, King Abdulaziz University
niksss (Singh, 2022)	1	Manipal University Jaipur

Table 3: Evaluation results of the best run for each team sorted by pRR

Team	Best Run	pRR	EM	$F_1@1$
TF200	TF200_run03	0.586	0.261	0.537
TCE (ElKomy and Sarhan, 2022)	MatMulMan_rejectAll	0.567	0.269	0.502
QQATeam (Ahmed et al., 2022)	QQATeam_Run02	0.559	0.244	0.513
GOF (Mostafa and Mohamed, 2022)	GOF_run01	0.546	0.239	0.525
stars (Wasfey et al., 2022)	stars_run06	0.528	0.256	0.507
DTW (Premasiri et al., 2022)	DTW_04	0.495	0.227	0.476
LK2022 (Alsaleh et al., 2022)	LK2022_run22	0.445	0.160	0.418
LARSA22 (Mellah et al., 2022)	LARSA22_run02	0.430	0.197	0.399
Rootroo	Rootroo_run03	0.409	0.092	0.364
SMASH (Keleg and Magdy, 2022)	SMASH_run03	0.400	0.151	0.382
eRock (Aftab and Malik, 2022)	eRock_testrun03	0.308	0.088	0.268
UM6P	NLPUM6P_run01	0.249	0.000	0.218
niksss (Singh, 2022)	niksss_run01	0.191	0.042	0.091

Pre-training transformer-based Language models trends. As expected, all of the 30 systems of the submitted runs leveraged variants of pre-trained transformer-based language models (LMs), with the majority using an encoder-only BERT-based model architecture. In contrast, only the LARSA22 team (Mellah et al., 2022) used a multilingual T5 (or mT5) encoder-decoder model architecture (Xue et al., 2021). Although such an architecture intrinsically supports sequence-to-sequence generative rather than extractive QA and MRC tasks, the best performing run for this team attained a pRR score of 0.430, which is very close to the median of all the pRR scores for the 30 runs in Table 4.

Naturally, the Arabic language was the main constituent of the dataset(s) used in pre-training the 30 models, 20 of which were pre-trained using MSA-only resources, and the remaining 10 were pre-trained using either multilingual resources, CA-only resources, or a mix of MSA, CA, and dialectal Arabic (DA) resources. Surprisingly, none of the LMs pre-trained using CA resources (exclusively or partially) have their respective systems/runs achieve pRR scores above the median (0.4375) of all pRR scores in Table 4). In Table 5, we list 2 runs of the SMASH team and two runs of the DTW team that belong to systems whose LMs were pre-trained using CA resources (exclusively as the case for the SMASH runs, and partially as the case of the DTW runs). All the attained scores are below the median. This is counter-indicative given that the Qur'an is in Classical Arabic. We speculate that adopting pre-trained models using CA-only resources or CA-resources combined with DA resources would prohibit or impede chances of transfer learning from

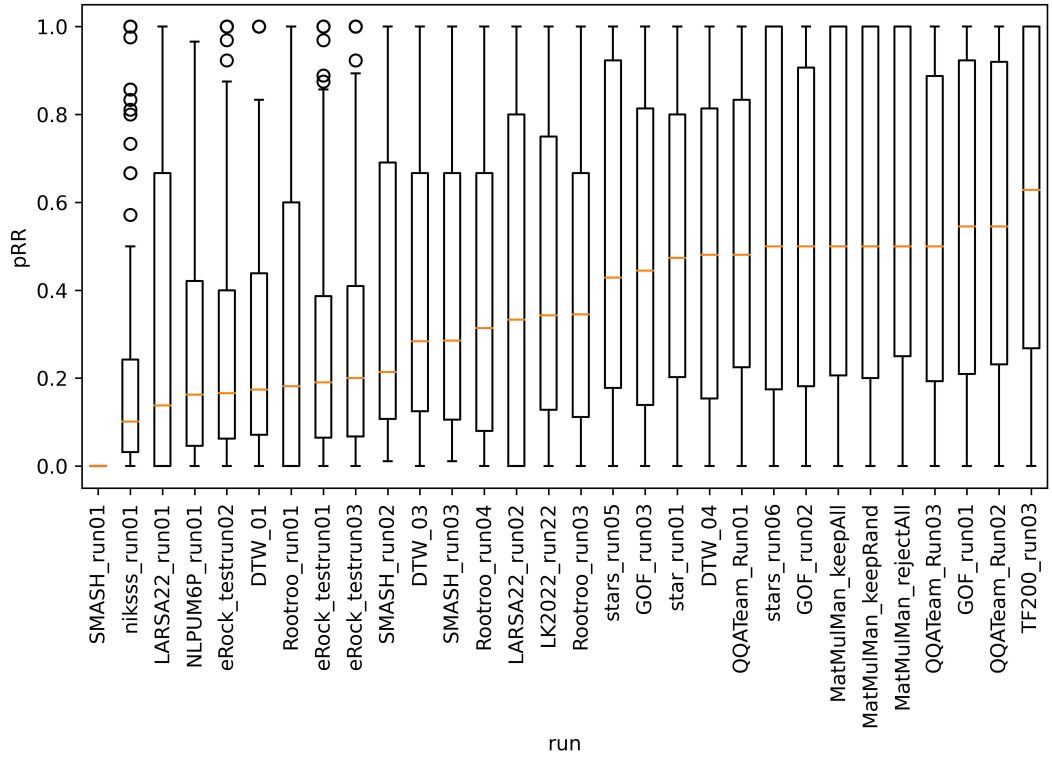

Figure 3: Boxplots for pRR metric for all submitted runs for Qur'an QA 2022 task. The plot illustrates the median and inter-quartile distance across questions.

MSA to CA. Albeit, this is needed given that the questions are in MSA and the answers are in CA.

Interestingly, only three of the 30 systems further pre-trained their language models using CA resources in an attempt to make them a better fit for the Qur'an QA task. The Rootroo team further pre-trained two multilingual BERT (mBERT) models (Devlin et al., 2019) using their crawled large corpus of Islamic and Fatwa websites, in addition to the verses of the Holy Qur'an. Whereas the Stars team (Wasfey et al., 2022) further pre-trained an AraBERT model using only the verses of Qur'an for their relatively least performing run (among their other two runs). This is not expected to make a significant improvement due to the relatively modest size of the Holy Qur'an to be used as the only CA resource in pre-training. Also, the performance of the submitted runs of the Rootroo team remained below the median (0.4375) of pRR scores in (Table 4), which may question the feasibility of further pre-training multilingual rather than monolingual MSA-only pre-trained models. This is a path worth further exploring.

Fine-tuning pre-trained language models trends. With respect to fine-tuning, all the systems used the *QRCD* training dataset in fine-tuning their pre-trained language models, either exclusively or in a pipelined fine-tuning procedure, where other MRC datasets were used in fine-tuning prior to using *QRCD*. The 10 systems/runs of the following 5 teams only used *QRCD* in fine-tuning: TCE, LK2022, LARSA22, niksss, and SMASH (Table 4). Except for the TCE team (ElKomy and Sarhan, 2022), none of these teams had runs with pRR scores exceeding the median (though LK2022_run22's pRR is almost equal to the median). In contrast, the three runs of the TCE team ranked second, fourth, and fifth, respectively, with respect to their pRR scores. However, the systems of these runs used variant combinations of effective post-processing schemes to improve their predicted answers. The excelling results of these runs may have out shadowed the importance of using large MRC datasets in a pipelined fine-tuning procedure, such as that adopted by the top performing run/system of the TF200 team. The latter team achieved the highest pRR score of **0.586** among all runs (Tables 3 and 4). Their system simply leveraged the MSA-only pre-trained AraELECTRA by Antoun et al. (2021) that was fine-tuned using the Arabic subset of the multilingual TyDiQA dataset (Clark et al., 2020), which they further fine-tuned using the *QRCD* dataset. Similarly, the QQATeam team (Ahmed et al., 2022) adopted a hybrid of the two MSA-only pre-trained AraELECTRA models, namely, AraELECTRA-ArTyDiQA and AraELECTRA-ARCD. The latter model was fine-tuned using the Arabic-SQuAD and the ARCD datasets by Mozannar et al. (2019) prior to fine-tuning using *QRCD*. They also adopted a data augmentation ap-

Table 4: Performance of all submitted runs ranked by pRR metric. Underlined rows are median runs.

Team	**Run**	pRR	EM	$F_1@1$
TF200	TF200_run03	**0.586**	0.261	**0.537**
TCE	MatMulMan_rejectAll	0.567	0.269	0.502
QQATeam	QQATeam_Run02	0.559	0.244	0.513
TCE	MatMulMan_keepAll	0.557	0.269	0.486
TCE	MatMulMan_keepRand	0.548	**0.273**	0.473
GOF	GOF_run01	0.546	0.239	0.525
QQATeam	QQATeam_Run03	0.535	0.231	0.500
Stars	stars_run06	0.528	0.256	0.507
QQATeam	QQATeam_Run01	0.526	0.223	0.462
stars	stars_run05	0.522	0.248	0.497
GOF	GOF_run02	0.521	0.244	0.501
stars	star_run01	0.502	0.181	0.483
DTW	DTW_04	0.495	0.227	0.476
GOF	GOF_run03	0.485	0.210	0.466
LK2022	LK2022_run22	0.445	0.160	0.418
LARSA22	LARSA22_run02	0.430	0.197	0.399
Rootroo	Rootroo_run03	0.409	0.092	0.364
DTW	DTW_03	0.408	0.139	0.390
SMASH	SMASH_run03	0.400	0.151	0.382
Rootroo	Rootroo_run04	0.392	0.113	0.354
SMASH	SMASH_run02	0.380	0.134	0.359
Rootroo	Rootroo_run01	0.323	0.113	0.282
LARSA22	LARSA22_run01	0.323	0.130	0.290
eRock	eRock_testrun03	0.308	0.088	0.268
DTW	DTW_01	0.290	0.084	0.258
eRock	eRock_testrun01	0.287	0.076	0.268
eRock	eRock_testrun02	0.280	0.076	0.246
UM6P	NLPUM6P_run01	0.249	0.000	0.218
niksss	niksss_run01	0.191	0.042	0.091
SMASH	SMASH_run01	0.000	0.000	0.000

Table 5: Runs of teams with language models pre-trained using CA resources exclusively (CAMeLBERT-CA) or partially (CAMeLBERT-Mix). The suffix "Mix" denotes MSA, CA and DA resources.

Team	Runs	Pre-trained Language Model	pRR
SMASH	SMASH_run03	CAMeLBERT-CA	0.400
SMASH	SMASH_run02	CAMeLBERT-CA	0.380
DTW	DTW_03	CAMelBERT-Mix	0.408
DTW	DTW_01	CAMelBERT-Mix	0.290

proach by paraphrasing the questions in *QRCD* to increase its size. The best run for this team ranked third with a pRR score of 0.559 3. Likewise, the GOF team adopted the same pipelined fine-tuning procedure and datasets employed by the QQATeam team, to have their best run rank fourth with a pRR score of 0.546 (Table 3). Mostafa and Mohamed (2022) experimented with two loss functions other than cross-entropy, one of which is a dynamically scaled cross-entropy loss that applies a modulating term to focus the learning process on low confidence examples (hard misclassified) and down-weigh the contribution of the high confidence examples (easy classified).

Among the multilingual MRC datasets that were used in fine-tuning are the MLQA (Lewis et al., 2020) and the XSQuAD (Artetxe et al., 2020) datasets. The Rootroo team fine-tuned their further pre-trained multilingual models (mentioned above) using the latter two datasets, in addition to the English SQuAD dataset. Thus, persisting in their attempts to explore the multilingual path they adopted.

Lastly, we describe the independent attempts by the Stars and the eRock teams to augment the *QRCD* training dataset prior to fine-tuning their respective models. Wasfey et al. (2022) and Aftab and Malik (2022) used the Annotated Corpus of Arabic Al-Qur'an Question and Answer (AQQAC) (Alqahtani and Atwell, 2018). The dataset is composed of 1,224 publicly available

QA pairs (in addition to 1000 unpublished ones) that have natural language answers *generated* from a Tafseer book, with each accompanied by its respective verse-based answer. Both teams were able to select and exploit about 500-740 questions. Questions were selected only if their respective answers could be extracted from the accompanying verse-based answer. For each selected QA pair, a context passage was generated for its verse-based answer from the Qur'an such that it matched the format of the *QRCD* dataset. Using the augmented dataset, the best performing run of the Stars team ranked fifth with a pRR score of 0.528 (well above the median), while the best run for the eRock team was much lower (0.308). The difference in performance could be mainly attributed to eRock's use of only the *QRCD* and the augmented dataset to fine-tune an ArabicBERT model (Safaya et al., 2020), as opposed to an AraBERT model (Antoun et al., 2020) that was fine-tuned using additional MSA MRC datasets (the Arabic SQuAd and ARCD) by the Stars team.

Ensemble Learning Trends. From a machine learning perspective, ensemble learning is regarded as the wisdom of the crowd, where multiple models vote towards a prediction (Sagi and Rokach, 2018). Three teams adopted an ensemble approach, namely, the TCE, Stars, and DTW, with the best performing run of the TCE team ranking second with a score of 0.567 (Table 3). They used an ensemble of three pre-trained language models, namely, AraBERTv0.2-Base and AraBERTv0.2-Large by Antoun et al. (2021) in addition to ARBERT (Abdul-Mageed et al., 2021), to vote for answer span predictions. Interestingly, we note that all three models are MSA-only pre-trained models; and models pre-trained using a mix of DA and MSA were excluded from their final runs. This may suggest (as we speculated above) that using Dialect Arabic may impede effective transfer learning from MSA to CA (though further exploration is needed). Similarly, the Stars team also used an ensemble of MSA-only pre-trained models in the system of their second submitted run, which attained a score of 0.522 (well above the median as shown in Table4).

In contrast, the DTW team adopted a self-ensemble approach to address the limitation of transformer models being prone to random seed initialization that may cause prediction fluctuations. As such, they trained their models using different random seeds and ensembled the prediction results over those models. Although self-ensemble may have marginally degraded their results, they still used it to ensure more stable predictions. Their best-performing run attained a score of 0.495.

7. Conclusion

The resurgence of the machine reading comprehension (MRC) field and the recent advances in intelligent deep learning models have motivated the organization of the first Qur'an Question Answering shared task, Qur'an QA 2022. The task aims at promoting state-of-the-art research on Arabic QA in general and MRC in particular on the Holy Qur'an.

We consider the first year of the shared task a big success, as it attracted 13 teams from 21 different institutes to participate in the final phase, with a total of 30 submitted runs. That clearly shows a relatively strong interest from the research community, despite the narrow domain of the task and its first-ever offering.

We have shed light on the main approaches adopted by the participating teams in the context of highlighting some of our perceptions and general trends of those approaches. All teams leveraged variants of pre-trained transformer-based language models. The majority used AraBERT and AraELECTRA, which were both pre-trained using MSA-only resources. The three top performing teams used AraELECTRA that was fine-tuned using large MRC datasets in MSA prior to fine-tuning using the *QRCD* dataset. We note that the second best team used an ensemble of two MSA-only pre-trained AraBERT models as well as an AraELECTRA model.

Our prospects towards the next version of the Qur'an QA shared task is to entail more challenging tasks that will attract a larger number of participants. A potential task is a QA task that requires systems to return answers from a given Surah or even the entire Qur'an.

8. References

Abdul-Mageed, M., Elmadany, A., et al. (2021). Arbert & marbert: Deep bidirectional transformers for arabic. In *Proceedings of the 59th Annual Meeting of the Association for Computational Linguistics and the 11th International Joint Conference on Natural Language Processing (Volume 1: Long Papers)*, pages 7088–7105.

Aftab, E. and Malik, M. K. (2022). eRock at Qur'an QA 2022: Contemporary Deep Neural Networks for Qur'an based Reading Comprehension Question Answers. 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)*.

Ahmed, B. H., Saad, M. K., and Refaee, E. A. (2022). QQATeam at Quran QA 2022: Fine-Tunning Arabic QA Models for Quran QA Task. 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)*.

Al-Azami, M. (2020). *The History of the Qur'anic Text from Revelation to Compilation: A Comparative Study with the Old and New Testaments, 2nd ed.* Turath Publishing, UK.

Alqahtani, M. and Atwell, E. (2018). Annotated corpus of Arabic al-quran question and answer.

Alsaleh, A., Althabiti, S., Alshammari, I., Alnefaie, S., Alowaidi, S., Alsaqer, A., Atwell, E., Altahhan, A.,

and Alsalka, M. A. (2022). LK2022 at Qur'an QA 2022: Simple Transformers Model for Finding Answers to Questions from Qur'an. 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)*.

Antoun, W., Baly, F., and Hajj, H. (2020). Arabert: Transformer-based model for arabic language understanding. In *LREC 2020 Workshop Language Resources and Evaluation Conference 11–16 May 2020*, page 9.

Antoun, W., Baly, F., and Hajj, H. (2021). Araelectra: Pre-training text discriminators for arabic language understanding. In *Proceedings of the Sixth Arabic Natural Language Processing Workshop*, pages 191–195. Association for Computational Linguistics.

Artetxe, M., Ruder, S., and Yogatama, D. (2020). On the cross-lingual transferability of monolingual representations. In *Proceedings of the 58th Annual Meeting of the Association for Computational Linguistics*, pages 4623–4637. Association for Computational Linguistics, July.

Atwell, E., Brierley, C., Dukes, K., Sawalha, M., and Sharaf, A.-B. (2011). An artificial intelligence approach to arabic and islamic content on the internet. In *Proceedings of NITS 3rd National Information Technology Symposium*, pages 1–8. Leeds.

Bashir, M. H., M. Azmi, A., Nawaz, H., Zaghouani, W., Diab, M., Al-Fuqaha, A., and Qadir, J. (2021). Arabic natural language processing for qur'anic research: A systematic review. April.

Chen, D., Fisch, A., Weston, J., and Bordes, A. (2017). Reading wikipedia to answer open-domain questions. In *Proceedings of the 55th Annual Meeting of the Association for Computational Linguistics (Volume 1: Long Papers)*, page 1870–1879. Association for Computational Linguistics.

Chen, D. (2018). *Neural Reading Comprehension and Beyond*. Ph.D. thesis, Stanford University.

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*, 8:454–470.

Devlin, J., Chang, M.-W., Lee, K., and Toutanova, K. (2019). Bert: Pre-training of deep bidirectional transformers for language understanding. 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)*, page 4171–4186. Association for Computational Linguistics, June.

ElKomy, M. and Sarhan, A. M. (2022). TCE at Qur'an QA 2022: Arabic Language Question Answering Over Holy Qur'an Using a Post-Processed Ensemble of BERT-based Models. 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)*.

Joshi, M., Choi, E., Weld, D., and Zettlemoyer, L. (2017). Triviaqa: A large scale distantly supervised challenge dataset for reading comprehension. In *Proceedings of the 55th Annual Meeting of the Association for Computational Linguistics (Volume 1: Long Papers)*, page 1601–1611. Association for Computational Linguistics.

Keleg, A. and Magdy, W. (2022). Smash at qur'an qa 2022: Creating better faithful data splits for low-resourced question answering scenarios. 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)*.

Lewis, P., Oguz, B., Rinott, R., Riedel, S., and Schwenk, H. (2020). MLQA: Evaluating cross-lingual extractive question answering. In *Proceedings of the 58th Annual Meeting of the Association for Computational Linguistics*, pages 7315–7330. Association for Computational Linguistics, July.

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*, 19(6):1–21, Nov.

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

Mellah, Y., Touahri, I., Kaddari, Z., Haja, Z., Berrich, J., and Bouchentouf, T. (2022). LARSA22 at Qur'an QA 2022: Text-to-Text Transformer for Finding Answers to Questions from Qur'an. 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)*.

Mostafa, A. and Mohamed, O. (2022). GOF at Qur'an QA 2022: Towards an Efficient Question Answering For The Holy Qu'ran In The Arabic Language Using Deep Learning-Based Approach. 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)*.

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

Premasiri, D., Ranasinghe, T., Zaghouani, W., Mitkov, R., Berrich, J., and Bouchentouf, T. (2022). DTW at Qur'an QA 2022: Utilising Transfer Learning with Transformers for Question Answering in a Low-resource Domain . In *Proceedings of the 5th Work-*

shop on Open-Source Arabic Corpora and Processing Tools (OSACT5) at the 13th Language Resources and Evaluation Conference (LREC 2022).

Rajpurkar, P., Zhang, J., Lopyrev, K., and Liang, P. (2016). Squad: 100, 000+ questions for machine comprehension of text. In *EMNLP*.

Safaya, A., Abdullatif, M., and Yuret, D. (2020). Kuisail at semeval-2020 task 12: Bert-cnn for offensive speech identification in social media. In *Proceedings of the Fourteenth Workshop on Semantic Evaluation*, pages 2054–2059.

Sagi, O. and Rokach, L. (2018). Ensemble learning: A survey. *Wiley Interdisciplinary Reviews: Data Mining and Knowledge Discovery*, 8(4):e1249.

Singh, N. (2022). niksss at Qur'an QA 2022: A Heavily Optimized BERT Based Model for Answering Questions from the Holy Qu'ran. 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)*.

Wasfey, A., Elrefai, E., Marwa, M., and Haq, N. (2022). Stars at Qur'an QA 2022: Building Automatic Extractive Question Answering Systems for the Holy Qur'an with Transformer Models and Releasing a New Dataset. 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)*.

Xue, L., Constant, N., Roberts, A., Kale, M., Al-Rfou, R., Siddhant, A., Barua, A., and Raffel, C. (2021). mt5: A massively multilingual pre-trained text-to-text transformer. In *NAACL-HLT*.