ParSQuAD: ***Persian Question Answering Dataset based on Machine Translation of SQuAD 2.0***

Negin Abadani[a], Jamshid Mozafari[a], Afsaneh Fatemi*[a], Mohammdali Nematbakhsh[a], Arefeh Kazemi[b]
[a]*Department of Software Engineering, University of Isfahan, Isfahan, Iran*
[b]*Department of Linguistics, University of Isfahan, Isfahan, Iran*
{negin.abadani, mozafari.jamshid}@gmail.com, {a_fatemi@, nematbakhsh}@eng.ui.ac.ir, arefeh_kazemi@yahoo.com

Received: 2021/07/02 Revised: 2021/10/16 Accepted: 2021/11/01

***Abstract*— Recent developments in Question Answering (QA) have improved state-of-the-art results, and various datasets have been released for this task. Since substantial English training datasets are available for this task, the majority of works published are for English Question Answering. However, due to the lack of Persian datasets, less research has been done on the latter language, making comparisons difficult. This paper introduces the Persian Question Answering Dataset (ParSQuAD) based on the machine translation of the SQuAD 2.0 dataset. Many errors have been discovered within the process of translating the dataset; therefore, two versions of ParSQuAD have been generated depending on whether these errors have been corrected manually or automatically. As a result, the first large-scale QA training resource for Persian has been generated. In addition, we trained three baseline models, i.e., BERT, ALBERT, and Multilingual-BERT (mBERT), on both versions of ParSQuAD. mBERT achieves scores of 56.66% and 52.86% for F1 score and exact match ratio respectively on the test set with the first version and scores of 70.84% and 67.73% respectively with the second version. This model obtained the best results out of the three on each version of ParSQuAD.**

Keywords—Question Answering; Persian Machine Reading Comprehension; Persian Question Answering Dataset; SQuAD

1. Introduction

Open-domain Question Answering (OpenQA) is an important task within the Natural Language Processing (NLP) and Information Retrieval (IR) fields. It aims to answer a question posed in natural language based on large-scale unstructured documents [1]. This procedure is carried out in two stages;

1. Retrieving relevant paragraphs from relevant documents,
2. Identifying an answer span within the retrieved relevant paragraphs, which is referred to as Machine Reading Comprehension (MRC).

Machine Reading Comprehension is a task in which a machine interprets natural language and answers questions by reading a passage. Traditional OpenQA systems have evolved into a modern architecture by adopting neural MRC methods to extract the answer to a given question from the relevant document(s) [2]–[5].

Training neural models for QA tasks requires large-scale datasets containing essential data for the task; therefore, with the recent increase in the number of OpenQA research, specifically on techniques that integrate with neural Machine Reading Comprehension, the need to generate datasets has increased.

Various large-scale QA datasets have been released recently, including CNN/Daily Mail [6], MS MARCO [7], RACE [8], and SQuAD [9]. However, the majority of these datasets are designed for English QA, and there are fewer or no datasets available for other languages, such as Persian. In other words, to this date, no similar open-domain dataset has been generated for Persian QA.

Among these recently released English datasets, the Stanford Question Answering Dataset (SQuAD) [9] has been used in most recent QA works.

This dataset comes in two different versions and contains (c, q, a) triplets representing a context paragraph from Wikipedia articles, a question posed by crowdworkers, and the related answer(s). The answer is a segment of the corresponding passage; therefore, a number also comes with the answer, indicating the answer's start position in the context paragraph. This dataset is divided into training and development sets, each having 80% and 10% of the total instances, respectively.

As mentioned, the SQuAD dataset comes in two versions;

- The first version, SQuAD 1.1 [9], contains over 100,000 instances from 536 articles.
- In order to generate SQuAD 2.0 [10], the second version, over 50,000 unanswerable questions have been added to the previous version. These questions are highly similar to the corresponding context paragraph but have incorrect answers since they cannot be found in that paragraph.

The idea behind adding such questions to the dataset was to challenge the existing models, and train models to correctly indicate unanswerable questions and not guess inappropriate answers.

In order to generate a Persian dataset based on SQuAD as the most popular datasets recently released, we first translated the SQuAD 2.0 [10] training and development sets using the Google Translate neural machine translation (NMT) API [11], then selected those questions that their translated answer(s) matched a portion of the context and removed the rest of the instances. After further polishing and modifying the result, we created a Persian dataset for the Question Answering tasks. In this paper, we introduce two versions of our dataset that have

10.22133/ijwr.2021.293313.1101 20.1001.1.26454335.2021.4.1.5.1

been generated in different ways based on the modification methods used.

To summarise, the main contributions of our work are:

i) Defining two different methods for modifying the SQuAD 2.0 Persian translation and overcoming the translation errors;

ii) Creating ParSQuAD, the first large-scale Persian QA dataset;

iii) Validating our dataset by establishing the current state-of-the-art for QA systems.

The rest of the paper is organised as follows: In section 2, some related works on dataset generation and translation have been reviewed; in section 3, we have described our methods for generating both versions of our translated dataset; also, we have analysed the dataset and compared it with other similar datasets in section 4; finally, in section 5 we have evaluated our datasets.

2. RELATED WORKS

The MRC field has seen tremendous growth in the last decade, including an increase in the number of corpus and significant advances in methods. Several English datasets for the Reading Comprehension (RC) task have been released to date, one of which would be the Stanford Question Answering Dataset (SQuAD) [9]. SQuAD is a reading comprehension dataset consisting of over 100,000 answerable questions and over 50,000 unanswerable questions posed by crowdworkers on a set of Wikipedia articles. The unanswerable questions were designed to look similar to answerable ones. Answerable questions are present in both versions of the dataset, whereas unanswerable questions have been added to SQuAD 2.0 [10]. In other words, SQuAD 1.1 data has been combined with over 50,000 unanswerable questions written by crowdworkers in order to generate SQuAD 2.0. In this dataset, the answer to each question is a portion of the corresponding passage.

Although these datasets have helped with the development of language-specific Question Answering models for English, the lack of native language annotated datasets other than English is one of the problems in this field. Various approaches have been proposed to generate non-English QA datasets; These approaches can be divided into four categories, shown in Fig. 1.

Major efforts have recently been made to generate a native Reading Comprehension dataset for languages such as Korean [12], Russian [13], Chinese [14], and French [15]. These datasets have been annotated with crowdworkers and contain native language passages; therefore, models trained with these datasets have a better quality in comparison with automatically generated datasets. However, generating datasets with crowdsourced annotations requires a group of experts and is costly and time-consuming.

A more cost and time-efficient solution would be to leverage a Neural Machine Translation (NMT) to translate the English datasets into target languages and fine-tune the language model on the translated dataset. Carrino et al. [16] translated the SQuAD 1.1 dataset into Spanish and trained a multilingual model to answer Spanish questions.

In some other works, authors have used both of the previously described methods to generate a dataset for the target language. Mozanner et al. [17] proposed the Arabic Reading Comprehension Dataset (ARCD) in order to deal with the lack of Arabic QA datasets. Their dataset consists of 1,395 questions posed by crowdworkers on Wikipedia articles (ARCD) and a machine translation of the SQuAD Dataset (Arabic-SQuAD). Lee et al. [18] created the Korean Question Answering Dataset (K-QuAD) semi-automatically, by using the automatically translated SQuAD and a QA system bootstrapped on a small set of question-answer pairs.

An alternative approach, proposed in [19], [20], is to provide a cross-lingual evaluation benchmark in order to enhance the development of cross-lingual Question Answering models. These models, unlike multilingual models, can transfer to a target language without requiring training data in that language. The XQuAD dataset [19] contains 1,190 instances from the SQuAD 1.1 development set, which have been translated by professionals into ten different languages. The MLQA dataset [20] is a combination of over 12000 English instances and 5000 instances in six other languages. Note that neither of the two datasets includes Persian instances.

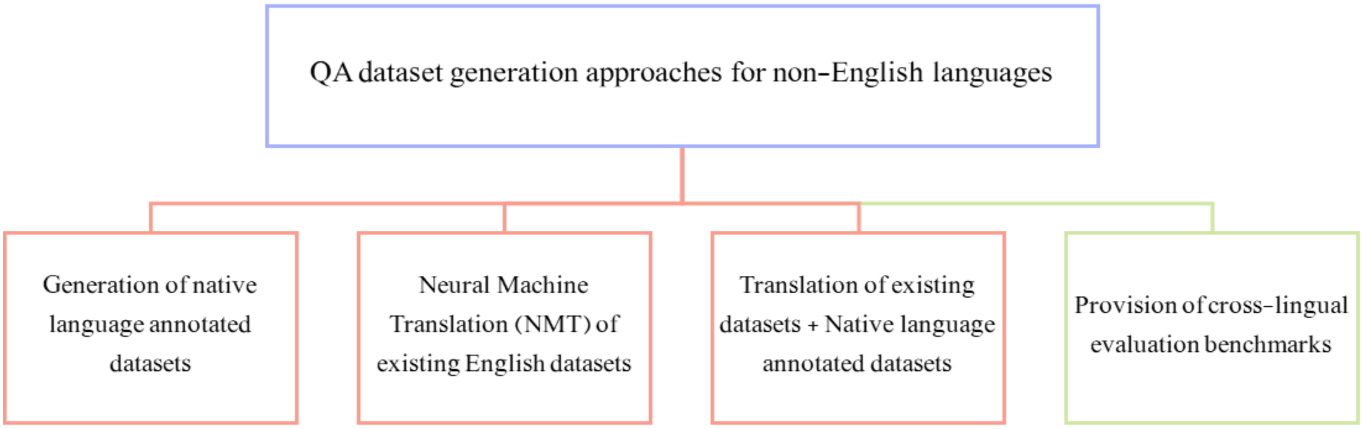

QA dataset generation approaches for non-English languages
Generation of native language annotated datasets
Neural Machine Translation (NMT) of existing English datasets
Translation of existing datasets + Native language annotated datasets
Provision of cross-lingual evaluation benchmarks

Fig. 1. QA dataset generation approaches for non-English languages; The first three approaches will lead to creating a new but monolingual data set, while the fourth approach will create multilingual datasets and train multilingual models.

In other cases, authors have decided to either use community-sourced datasets or develop close-domain QA systems. As for Persian QA, Tohidi et al. [21] employed the Rasekhoon question answering dataset (www.rasekhoon.net) to evaluate their question matching model. Veisi et al. [22] collected and structured a dataset of diseases and drugs to evaluate their Persian medical question answering system. Boreshban et al. [23] developed a religious Persian QA corpus, Rasayel&massayel. Also, TriviaQA [24], is a collection of question-answer pairs written by trivia fans, as well as independently acquired documents containing the answers to the questions.

In order to fill the gap for Persian QA, we generated a Persian machine-translated dataset (ParSQuAD) based on SQuAD 2.0. ParSQuAD comes in two versions; one of which has been manually modified, and no manual modifications have been done on the other version; in fact, it has been modified automatically. The first version of the ParSQuAD dataset contains over 25,000 questions, whereas the second version has over 70,000 questions. The development set has also been translated and modified into two different ways to suit translated training set versions respectively.

The modification process of the resulting datasets will be discussed in section 3.

3. DATASET GENERATION METHODOLOGY

In this section, the process of generating both versions of the ParSQuAD dataset will be explained. As mentioned in the previous section, this dataset is based on a machine translation of SQuAD 2.0 using the Google Translate neural machine translation (NMT) API [11].

Before explaining the procedure, we need to understand the structure of the SQuAD dataset. This dataset has a tree-like structure;

- It consists of several titles;
- Each title includes related passages;
- Each passage has two types of questions: answerable and unanswerable;

Each question has a set of answers with a corresponding number as the start span. Note that the start span indicates the start position of the answer in its corresponding passage. The structure can be seen in Fig. 2.

Based on the dataset's structure, we understand that it consists of triplets of context paragraph, answer, and question. In order to generate the Persian dataset, the first step is to translate all original (c_o, q_o, a_o) instances into (c_t, q_t, a_t) triplets. The next step would be to change the start spans for each answer accordingly, but before that, we need to make sure that the translation has the highest possible quality or the best proportion of the translated dataset has been selected.

In terms of translation quality, the translator performed well, except for words containing Hamza. Hamza (ء) is an Arabic letter that can appear in different forms (ء, أ, إ, ؤ, ئ) and at any position in a word. The letter is also used in loanwords from Arabic in Persian since it is written in an Arabic-based alphabet. As mentioned, the Google Translate API occasionally misspells words containing a Hamza letter and replaces the letter with an English word from the original sentence.

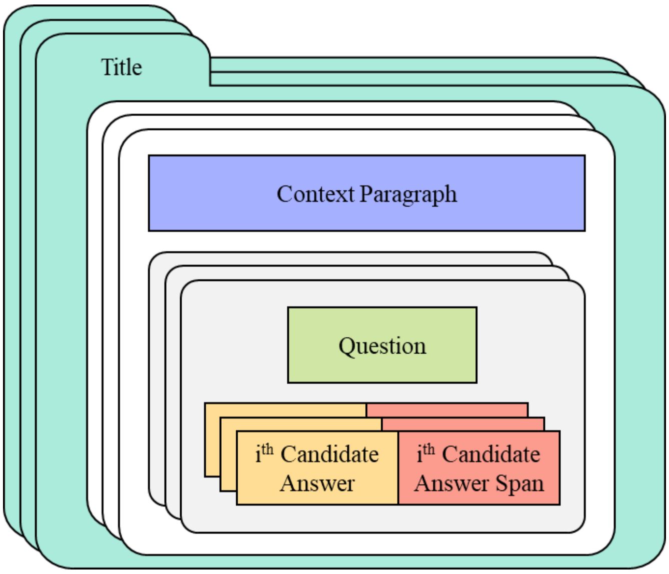


Fig. 2. The SQuAD dataset structure.

Fig.3 depicts different English passages and their Persian translation as examples of this behaviour; the API has functioned correctly in the green highlighted parts, whereas the parts highlighted in red are examples of where it has failed to spell the words correctly.

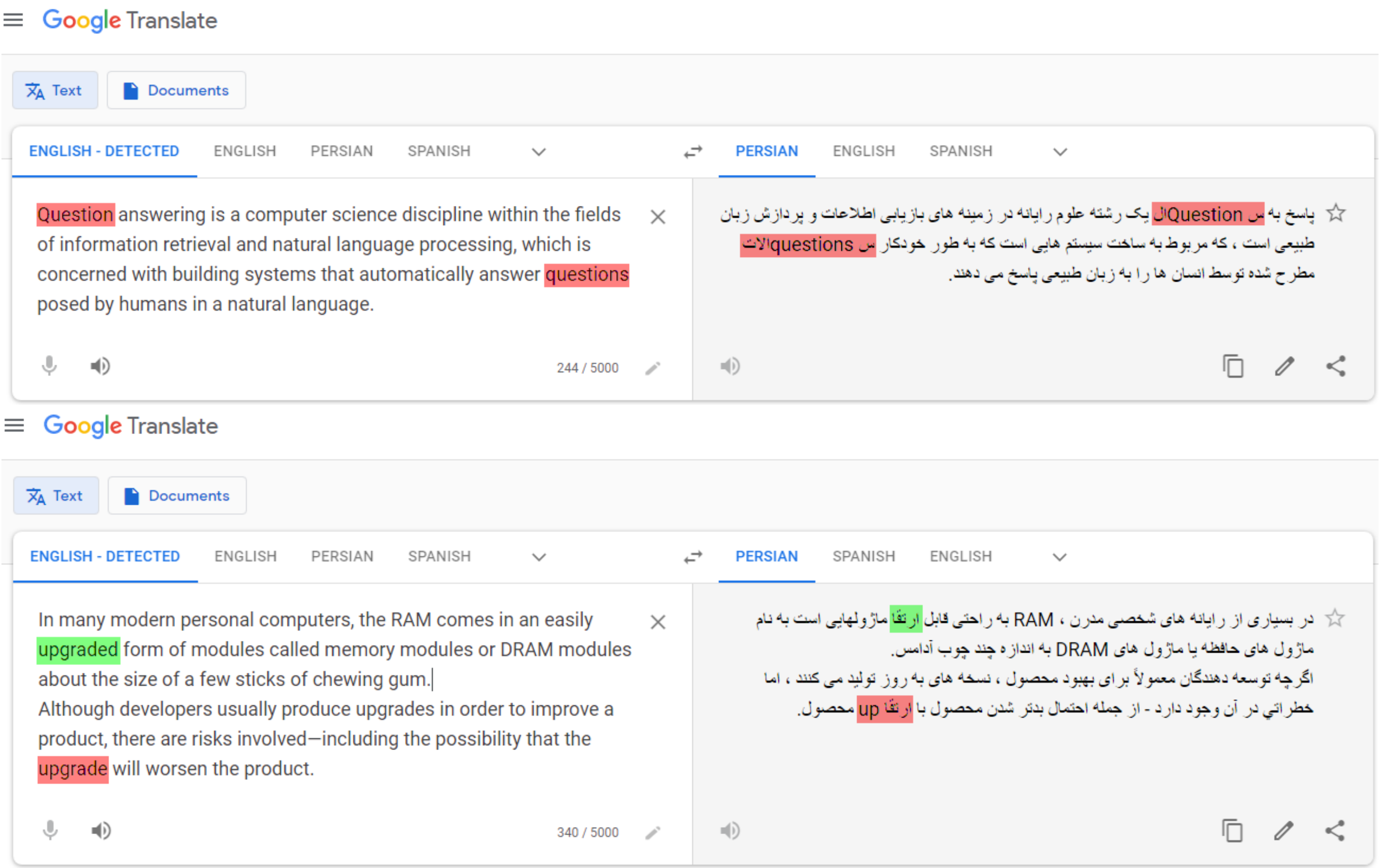
Since the dataset contains a large number of sentences, finding and correcting the Hamza translation errors would be time-consuming and manual correction leads to new mistakes due to human error.

After increasing the quality of the translated dataset, another challenge was to find the correct answer span for the translated version of each triplet. As expected, the structure of some sentences had changed after translation; therefore, the answer did not match the context paragraph. This issue makes those triplets unusable; therefore, those triplets have been removed.

Based on the approaches used to deal with the translation quality and answer start span correction issues, two different versions of ParSQuAD have been generated. The two methods used for dealing with these issues are correction or removal of the problematic passages. Since correction is complex for the second challenge, it has to be done manually. On the other hand, removing the problematic passages from the dataset is an easier alternative that can be done automatically through a predefined algorithm. Therefore it is possible to classify these two approaches into manual and automatic.

Note that since the number of Hamza translation errors was significant and manual corrections may have led to new mistakes due to possible human errors, these errors have been ignored in the manual method. The differences in both methods can be seen in Fig. 4.

Also, the automatic version is three times larger than the manual version. The main reason for the size difference is that the dataset was translated in batches. To overcome the limitations of the API, we divided the training dataset into 13 batches and translated each batch separately. After having



Question answering is a computer science discipline within the fields of information retrieval and natural language processing, which is concerned with building systems that automatically answer questions posed by humans in a natural language.

پاسخ به س Questionال یک رشته علوم رایانه در زمینه های بازیابی اطلاعات و پردازش زبان
طبیعی است ، که مربوط به ساخت سیستم هایی است که به طور خودکار س questionsالات
مطرح شده توسط انسان ها را به زبان طبیعی پاسخ می دهند.

In many modern personal computers, the RAM comes in an easily upgraded form of modules called memory modules or DRAM modules about the size of a few sticks of chewing gum.
Although developers usually produce upgrades in order to improve a product, there are risks involved—including the possibility that the upgrade will worsen the product.

در بسیاری از رایانه های شخصی مدرن ، RAM به راحتی قابل ارتقا ماژولهایی است به نام
ماژول های حافظه یا ماژول های DRAM به اندازه چند چوب آدامس.
اگرچه توسعه دهندگان معمولاً برای بهبود محصول ، نسخه های به روز تولید می کنند ، اما
خطراتي در آن وجود دارد - از جمله احتمال بدتر شدن محصول با ارتقا up محصول.

Fig. 3. Different English passages and their Persian translation obtained by using Google Translate neural machine translation (NMT) API. Parts highlighted in green show that the API has worked correctly, but the red highlighted parts show it has failed to correctly spell words containing Hamza letters.

translated the first four batches, we decided to modify these four translated batches further manually and use the result to train our models.

Note that, since the final form of this version had the minimum size required for training a model, we decided not to remove the errors and instead corrected them manually.

After successfully translating all 13 batches of the original dataset, we automatically removed the errors and generated the second version since there was already enough data to train a model. The two methods used for generating both versions of the dataset will be discussed in detail in the following two sections.

A. *The Manual Method*

As mentioned, since the first version of the translated dataset was smaller in size, it was easier and necessary to apply manual modifications to this version to avoid reducing the size any further. Therefore, both errors described have been manually corrected.

Since after translating the dataset, the structure of sentences had changed, the first step was to find those (q_t, a_t, c_t) triplets whose answer a_t perfectly matched a subset of the corresponding context paragraph c_t. After finding those triplets, the next step was to correct the start span of the answer a_t based on the context c_t. This raised another issue; if the answer has appeared multiple times in the context paragraph, how can the correct span be selected?

In order to avoid selecting the incorrect answer span when the answer appears in more than one sentence, the sentence in the original context the answer has appeared in had to be determined based on the original strat span. After finding the sentence number and mapping the number to the translated context paragraph in order to find the corresponding translated sentence, the new start span had to be located in the translated sentence.

Although this simple method would have solved the start span problem, but after evaluating the dataset, it appeared that some of the start spans did not point to the answer. It seemed that this problem had been caused by the presence of punctuation marks in the context paragraph, namely, single (‘ ’) and double (“ ”) quotation marks. In this case, since there was a small number of wrong start spans and no possible algorithmic solution for the issue, we decided to correct the remaining wrong start spans manually.

As for the Hamza translation errors, since there were many possible translation errors and finding those errors required a thorough study of the dataset, no action was taken on this version of the dataset to correct those errors.

It is important to note that before deciding to leave these Hamza translation errors as is, the first attempt was to correct them manually, which resulted in other errors that occurred due

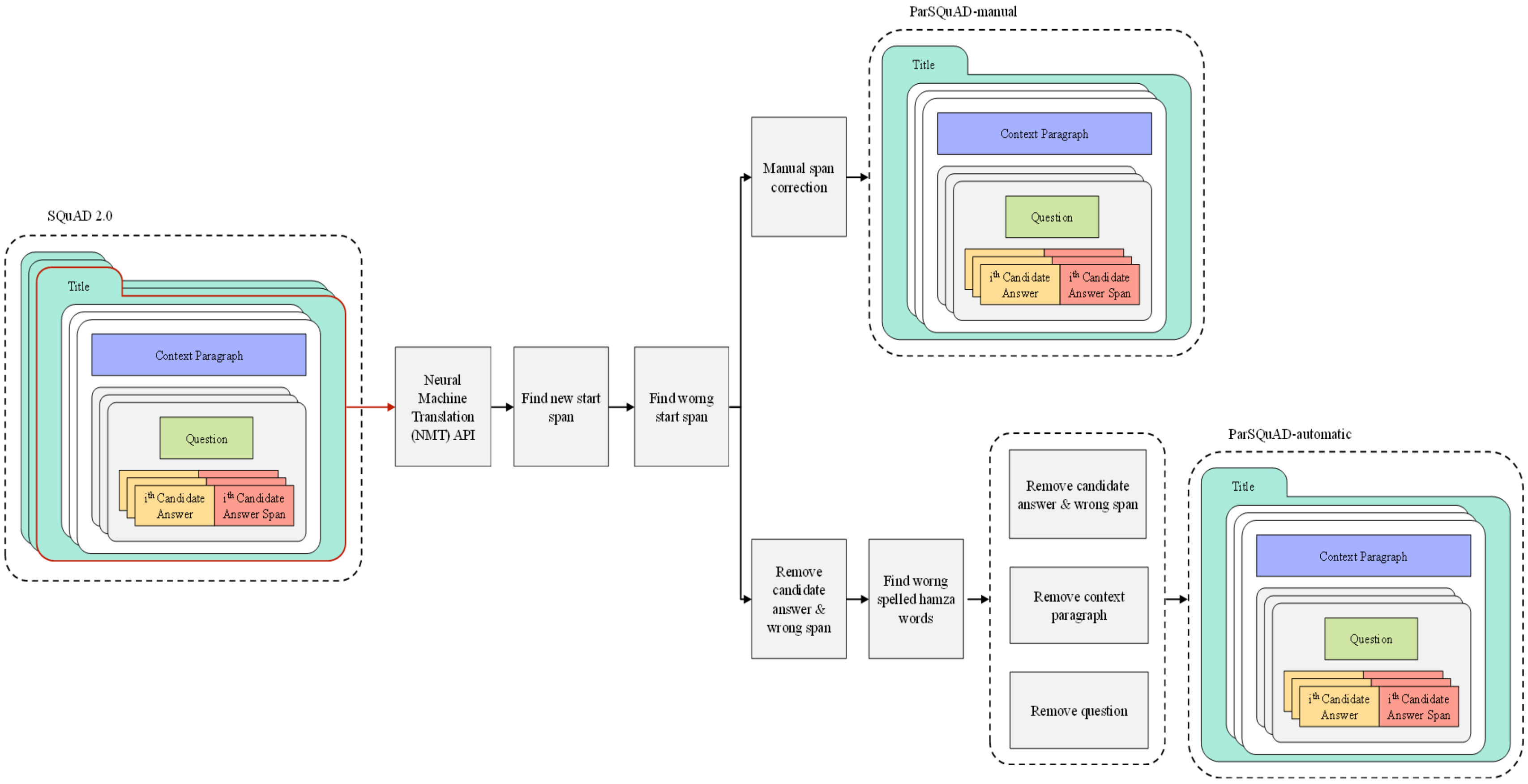

SQuAD 2.0
Title
Context Paragraph
Question
ith Candidate Answer
ith Candidate Answer Span
Neural Machine Translation (NMT) API
Find new start span
Find worng start span
Manual span correction
ParSQuAD-manual
Title
Context Paragraph
Question
ith Candidate Answer
ith Candidate Answer Span
Remove candidate answer & wrong span
Find worng spelled hamza words
Remove candidate answer & wrong span
Remove context paragraph
Remove question
ParSQuAD-automatic
Title
Context Paragraph
Question
ith Candidate Answer
ith Candidate Answer Span

Fig. 4. Summary of the two methods (i.e., Manual and Automatic) used in this paper to generate ParsQuAD datasets.

to possible human error. Algorithm 1 describes the manual method's implementation. Also, Fig. 5 shows its flowchart.

B. The Automatic Method

The second version of the dataset includes translated instances from all 13 batches of the SQuAD dataset; therefore, the number of errors that appeared in this version was larger. Since the size of this version was enough for training models, only automatic modifications have been applied to this version.

After extracting (q_t, a_t, c_t) triplets whose answer a_t perfectly matched a subset of the corresponding context paragraph c_t and correcting the start answer spans, any (q_t, a_t, c_t) triplets whose start span did not point to the answer were automatically removed from the dataset. Also, if any triplet contained translation errors, the triplet has been automatically removed entirely. Algorithm 2 shows the automatic method's implementation. Also, Fig. 6 visualises this algorithm.

4. DATASET ANALYSIS

To better understand the differences between the manual and automatic versions of ParSQuAD, their properties have been analysed in this section.

Table I summarises the statistics of the two final versions of the ParSQuAD dataset regarding the number of questions and titles translated compared to the total number of questions and titles in the original English dataset. In addition, Fig. 7 compares the datasets based on the percentage of answerable and unanswerable questions.

According to Fig. 7, the training set of the first version has fewer unanswerable questions than the number of answerable questions. Therefore, this version does not provide a fair representation of the original dataset. On the other hand, the second training set contains a higher percentage of unanswerable questions compared to the first version, making it a better representation of the original SQuAD 2.0 dataset. This percentage could affect the performance of the trained model, which will be discussed in section 5.

Due to three reasons, the automatic version of ParSQuAD is about half the size of the original SQuAD dataset;

First, the API sometimes failed to translate some of the passages. In order to resolve this, we tried sending those passages to the API multiple times. Even though this method helped with the issue, but in the end, a significant amount of untranslated passages remained.

Second, the structure of an English passage differs from that of Persian, hence, when translating a passage from English to Persian, the structure of each sentence might vary, but the structure of a sequence of words might not change. This leads to the translated answer not matching the respective context paragraph; therefore, the answer span cannot be found, and the (q_t, a_t, c_t) example will be removed from the translated dataset.

TABLE I. STATISTICS OF THE TWO VERSIONS OF PARSQUAD IN COMPARISON WITH THE SQUAD 2.0 DATASET

Dataset	**Training set**		**Development set**	
	# questions	*# titles*	*# questions*	*# titles*
SQuAD 2.0	130319	442	11873	35
ParSQuAD-manual	18906	136	5726	35
ParSQuAD-automatic	64961	442	5599	35

Algorithm 1. Implementation of the manual method.
c_o and c_t : the original and translated context,
q_o and q_t : the original and translated question,
a_o and a_t : the original and translated answer,
a_t^{start} : the translated answer start span.

```
Result: ($c_t$, $q_t$, $a_t$, $a_t^{start}$)
for $c_o$ in context paragraphs do
    $c_t \leftarrow$ get context translation;
    for $q_o$ in questions do
        for $a_o$ in answers do
            $a_t \leftarrow$ get answer translation;
            if $a_t$ in $c_t$ then
                compute original sentence number $a_o$ appeared in;
                $n \leftarrow$ get original sentence number;
                $s_t \leftarrow$ get n-th translated sentence;
                if $a_t$ in $s_t$ then
                    compute $a_t$ start span;
                    $a_t^{start} \leftarrow$ get translated answer start span;
                    return ($c_t$, $q_t$, $a_t$, $a_t^{start}$);
        end
    end
    if $a_t$ not in $c_t[a_t^{start} : c_t^{end}]$ then
        manually edit the $a_t^{start}$;
end
```

Algorithm 2. Implementation of the automatic method.
c_o and c_t : the original and translated context,
q_o and q_t : the original and translated question,
a_o and a_t : the original and translated answer,
a_t^{start} : the translated answer start span,
D : a dictionary of all Persian words containing Hamza letters,
E : a set of all English letters.

```
Result: ($c_t$, $q_t$, $a_t$, $a_t^{start}$)
$D \leftarrow$ get the dictionary of Persian words containing Hamza;
$E \leftarrow$ get English letters;
replace Hamza and tail-end in $H$ with each letter in $E$;
H $\leftarrow$ get equivalence classes of newly created words;
for $c_o$ in context paragraphs do
    $c_t \leftarrow$ get context translation;
    for $q_o$ in questions do
        for $a_o$ in answers do
            $a_t \leftarrow$ get answer translation;
            if $a_t$ in $c_t$ then
                compute original sentence number $a_o$ appeared in;
                $n \leftarrow$ get original sentence number;
                $s_t \leftarrow$ get n-th translated sentence;
                if $a_t$ in $s_t$ then
                    compute $a_t$ start span;
                    $a_t^{start} \leftarrow$ get translated answer start span;
                    return ($c_t$, $q_t$, $a_t$, $a_t^{start}$);
        end
    end
    if $a_t$ not in $c_t[a_t^{start} : c_t^{end}]$ then
        remove ($q_t$, $a_t$, $a_t^{start}$);
    for $h_i$ in $H=\{h_1, h_2, \dots, h_m\}$ do
        if $h_i$ in $c_t$ or $q_t$ or $a_t$ then
            remove ($c_t$, $q_t$, $a_t$, $a_t^{start}$);
    end
end
```

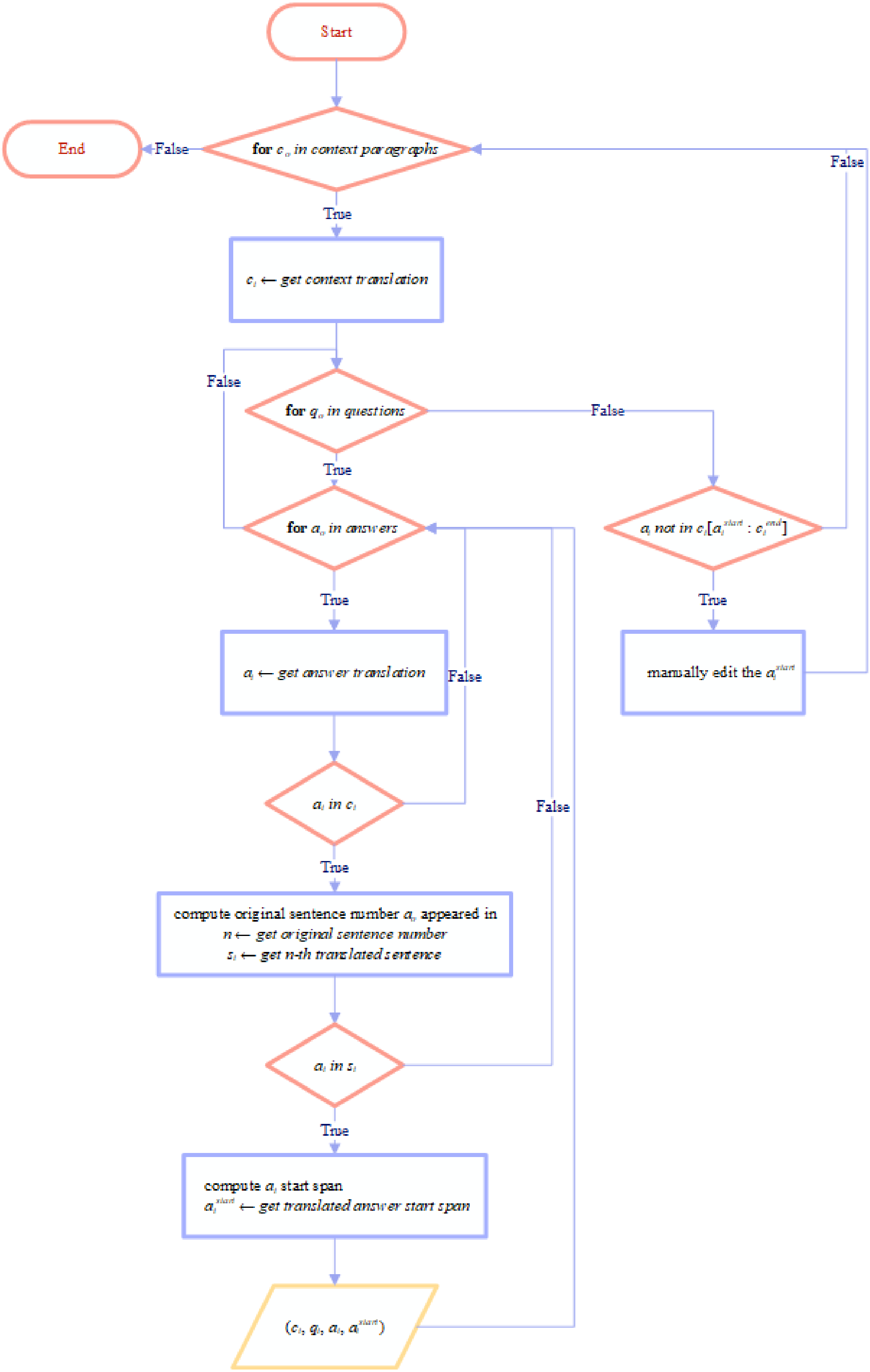

Start
End
False
for c_o in context paragraphs
False
True
$c_t \leftarrow$ get context translation
False
for q_o in questions
False
True
for a_o in answers
a_t not in $c_t[a_t^{start} : c_t^{end}]$
True
True
$a_t \leftarrow$ get answer translation
False
manually edit the a_t^{start}
a_t in c_t
False
True
compute original sentence number a_o appeared in
$n \leftarrow$ get original sentence number
$s_t \leftarrow$ get n-th translated sentence
a_t in s_t
True
compute a_t start span
$a_t^{start} \leftarrow$ get translated answer start span
$(c_t, q_t, a_t, a_t^{start})$

Fig. 5. The flowchart of the manual method described in Algorithm 1.

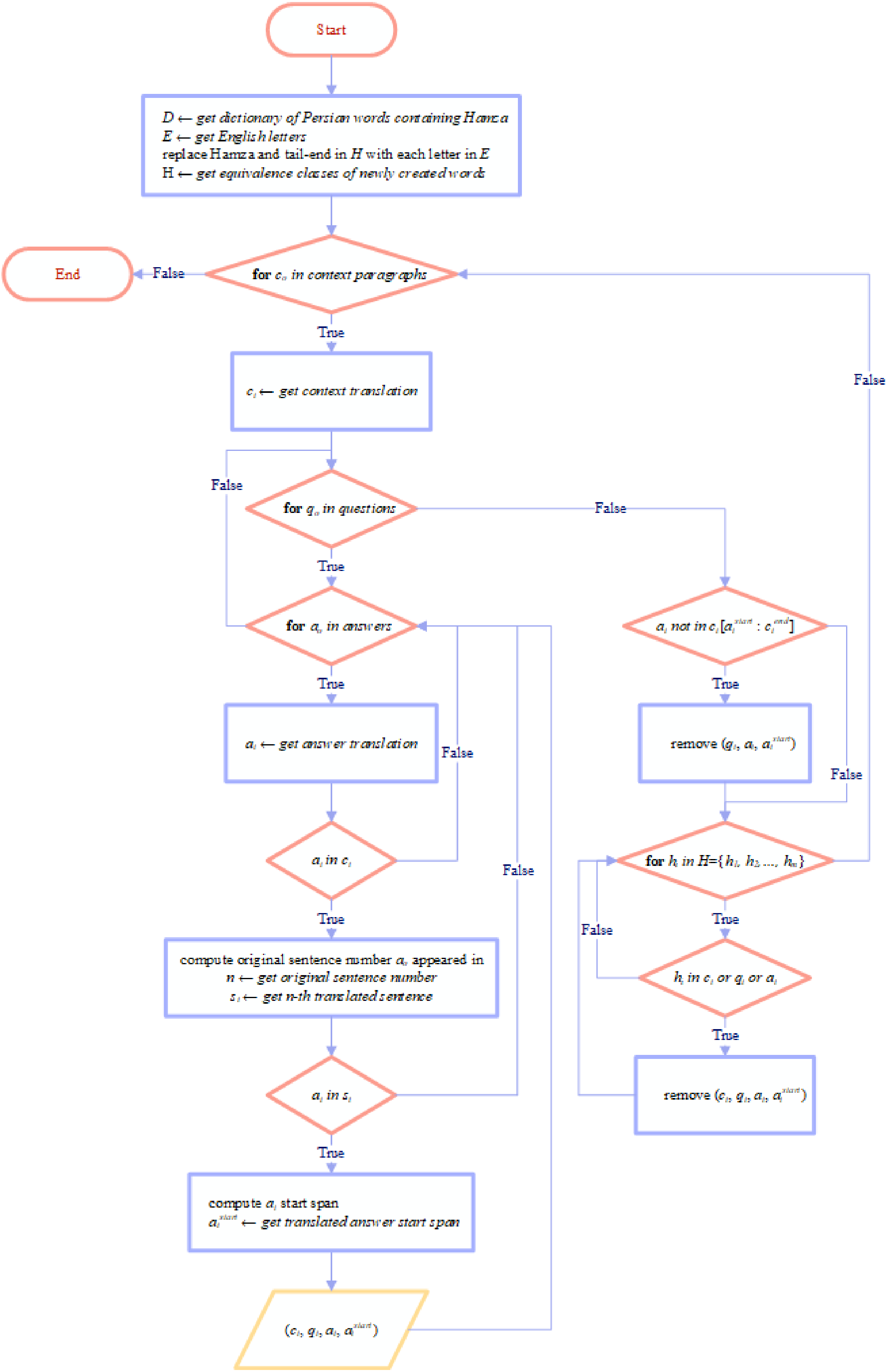

Start
$D \leftarrow$ get dictionary of Persian words containing Hamza
$E \leftarrow$ get English letters
replace Hamza and tail-end in H with each letter in E
$H \leftarrow$ get equivalence classes of newly created words
End
False
for c_o in context paragraphs
True
$c_i \leftarrow$ get context translation
False
False
for q_o in questions
False
True
for a_o in answers
a_i not in $c_i[a_i^{start} : c_i^{end}]$
True
True
$a_i \leftarrow$ get answer translation
False
remove (q_i, a_i, a_i^{start})
False
a_i in c_i
False
for h_i in $H=\{h_1, h_2, ..., h_m\}$
True
True
False
compute original sentence number a_o appeared in
$n \leftarrow$ get original sentence number
$s_i \leftarrow$ get n-th translated sentence
h_i in c_i or q_i or a_i
True
a_i in s_i
remove $(c_i, q_i, a_i, a_i^{start})$
True
compute a_i start span
$a_i^{start} \leftarrow$ get translated answer start span
$(c_i, q_i, a_i, a_i^{start})$

Fig. 6. The flowchart of the automatic method described in Algorithm 2.

Third, the translation errors described in previous sections resulted in eliminating many translated (q_t, a_t, c_t) examples from the translated dataset.

Furthermore, the charts in Fig. 8 compare the distribution of unanswerable questions in ParSQuAD-automatic and SQuAD 2.0. Each chart shows the frequency of different numbers of unanswerable questions available for a context paragraph. It can be observed that both datasets have a similar distribution of unanswerable questions. Note that about half of the paragraphs do not have unanswerable questions, which is due to the fact that the SQuAD 2.0. has been built upon SQuAD 1.1 by adding unanswerable questions to the latter dataset.

Table II summarises the English and non-English datasets generated for QA tasks that have been reviewed in section 2. This table compares our datasets with others in terms of contacting unanswerable questions, their sizes, method of generation, and language. Note that only those with the same structure as the SQuAD dataset have been included.

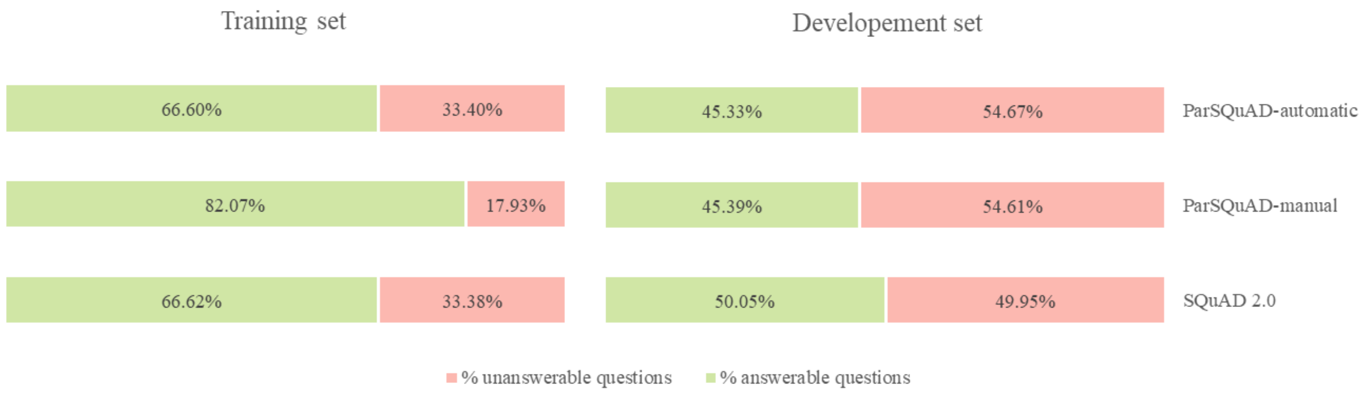
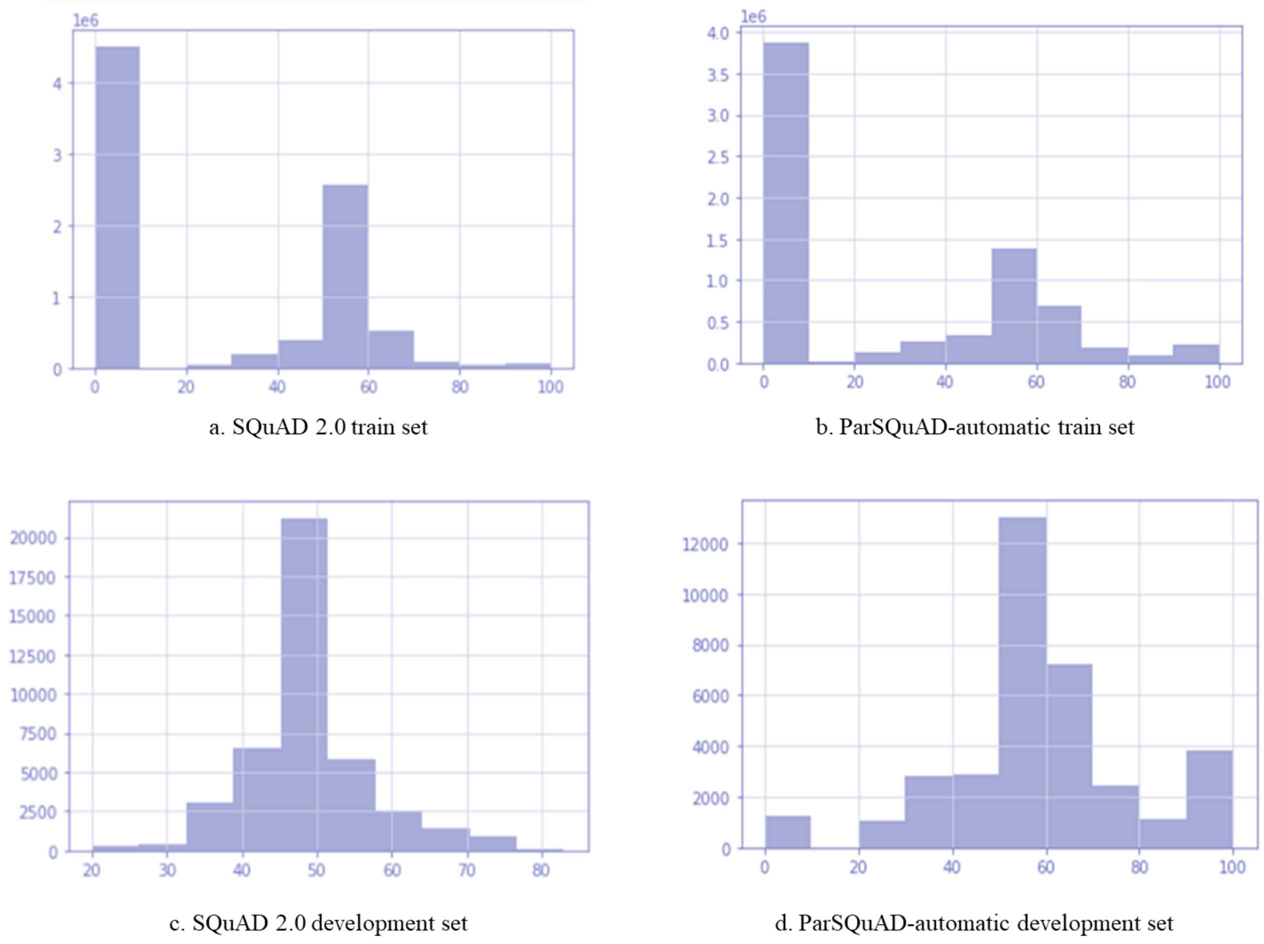


Fig. 7. Percentage of answered and unanswerable questions in the SQuAD 2.0 dataset and the two versions of ParSQuAD.



a. SQuAD 2.0 train set

b. ParSQuAD-automatic train set

c. SQuAD 2.0 development set

d. ParSQuAD-automatic development set

Fig. 8. Distribution of unanswerable questions in SQuAD 2.0 dataset and ParSQuAD-automatic.

TABLE II. Statistics of Available question answering datasets in comparison with the two versions of ParSQuAD

Dataset	Size (# questions)	Contains unanswerable questions	Type	Language
SQuAD 1.1 [9]	100,000	✗	Native	English
SQuAD 2.0 [10]	150,000	✓	Native	English
KorQuAD 1.0 [12]	70,000	✗	Native	Korean
SberQuAD [13]	90,000	✗	Native	Russian
Chinese Span-Extraction dataset [14]	20,000	✗	Native	Chinese
FQuAD 1.0 [15]	25,000	✗	Native	French
FQuAD 1.1 [15]	60,000	✗	Native	French
ARCD [17]	1,000	✗	Native	Arabic
SQuAD-es [16]	87,000	✗	Translation	Spanish
SQuAD-es-small [16]	46,000	✗	Translation	Spanish
Arabic-SQuAD [17]	48,000	✗	Translation	Arabic
Arabic-SQuAD + ARCD [17]	50,000	✗	Translation + Native	Arabic
K-QuAD [18]	81,000	✗	Translation + Native	Korean
XQuAD [19]	13,000	✗	Translation + Native	English, Spanish, German, Greek, Russian, Turkish, Arabic, Vietnamese, Thai, Chinese, Hindi
MLQA [20]	42,000	✗	Translation + Native	English, Arabic, German, Spanish, Hindi, Vietnamese, Chinese
TriviaQA [24]	95,000	✗	Community-sourced	English
ParSQuAD-manual	25,000	✓	Translation	Persian
ParSQuAD-automatic	70,000	✓	Translation	Persian

5. EXPERIMENTS

In order to evaluate and compare the quality of our datasets with the SQuAD 2.0, three QA models have been trained. The pre-trained BERT (ParsBERT [25] for the translated dataset), ALBERT, and Multilingual-BERT (mBERT) models have been fine-tuned on both versions of the ParSQuAD dataset and the SQuAD 2.0 to generate our three final Question Answering models. Note that another version of ALBERT pre-trained on Persian (ALBERT-Persian) has been used for the ParSQuAD dataset. Our models have been trained for two epochs using a Tesla P100-PCIE-16GB GPU device with the default parameter's values set in the HugginFace scripts.

In order to evaluate the accuracy of the trained models, similar to SQuAD, two measures have been employed;

Exact Match (EM): This metric is used to calculate the percentage of predictions that match the ground truth answer perfectly.

F-Measure (F1-Score): The maximum overlap between the predicted answer and the ground truth answer at the token level is measured by this metric.

The goal is to evaluate the quality of our datasets, which will be used as training resources for Persian QA models. Table III summarises the evaluation results for each dataset.

Also, the chart displayed in Fig. 9 is based on the data provided in Table III. This chart shows that ALBERT models work better for the SQuAD 2.0 dataset, while mBERT works best on both versions of the ParSQuAD dataset, achieving scores of 56.66% and 52.86% for F1 score and exact match ratio respectively on the test set with the first version and scores of 70.84% and 67.73 % respectively with the second version. This could be due to some words in languages other than Persian remaining in the translation; Therefore, a multilingual model would work better on a dataset containing such passages.

In addition, we observe that models trained on SQuAD 2.0 and ParSQuAD-automatic have a higher F1 score when compared to their exact match ratio; While it is the opposite for models trained on ParSQuAD-manual. This indicates that the ParSQuAD-manual dataset trained the models to predict the majority of the test questions as unanswerable, which proves that the models require more training on unanswerable questions and that the train set should include more unanswerable questions.

As discussed earlier in section 4, the first version of the dataset does not provide a fair representation of the original dataset. In other words, this version does not reflect the original SQuAD dataset in terms of the proportion of answerable and unanswerable questions. Therefore, in order to study the effect of this ratio on the accuracy of the trained model, we added 3,708 unanswerable questions from new titles to the manual version of ParSQuAD, doubling the number of such questions. This increased the ratio of unanswerable questions to 28.01%. After training all three models on the resulting dataset, the scores improved by 3% on average, demonstrating that the ratio of unanswerable questions in the dataset affects the accuracy of the trained model. Fig. 10 shows a summary of the results.

TABLE III. EVALUATION RESULTS FOR EACH DATASET ON THREE DIFFERENT MODELS.

Dataset	**SQuAD 2.0**		**ParSQuAD-manual**		**ParSQuAD-automatic**	
Model	*EM*[b]	*F1*[c]	*EM*	*F1*	*EM*	*F1*
BERT[a]	72.74	75.86	46.32	50.06	62.42	65.26
ALBERT	78.98	82.15	48.11	51.66	64.71	67.59
mBERT	74.92	78.09	52.86	56.66	67.73	70.84

[a] ParsBERT has been used for ParSQuAD.

[b] Exact Match

[c] F1: F-one measure

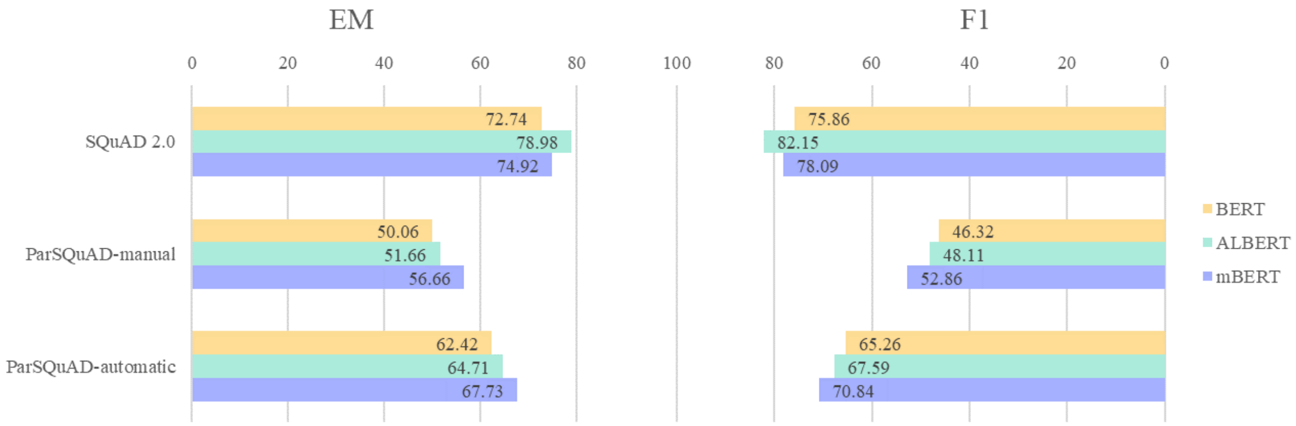


Fig. 9. Exact match ratio and F1 score for the three models trained on each dataset.

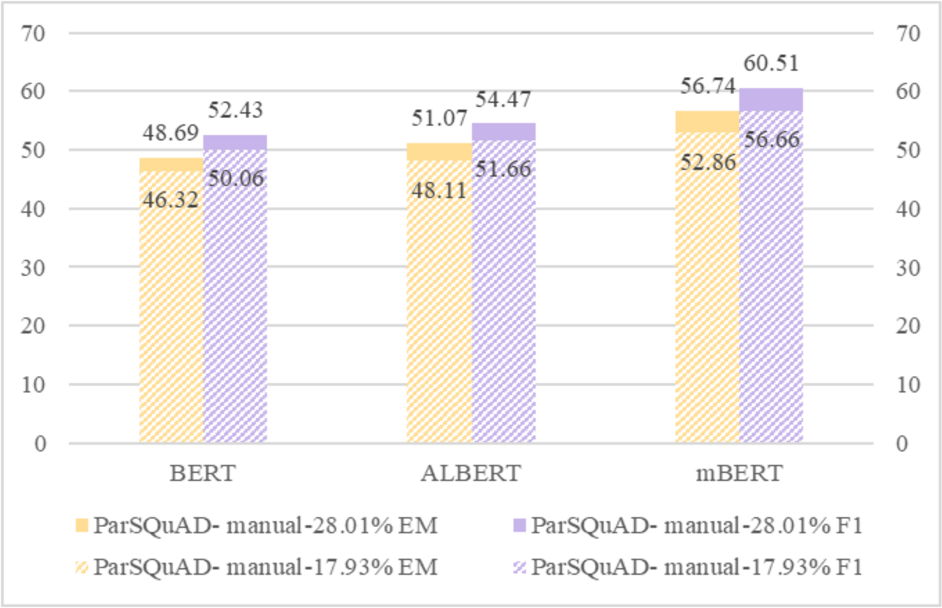


Fig. 10. A comparison of the evaluation results for two versions of ParSQuAD-manual with different sizes on three different models.

The 3% improvement in the scores proves that encountering a certain type of question more frequently may improve the learning process; by increasing the number of unanswerable questions, the model learned and detected these questions better.

6. CONCLUSION

This paper introduced a Persian dataset generated from the SQuAD 2.0 dataset for Persian Machine Reading Comprehension. To the best of our knowledge, it is the first dataset for Persian Reading Comprehension based on the structure of the SQuAD dataset.

With the help of the Google Translate neural machine translation (NMT) API, a translated version of the SQuAD 2.0 dataset has been generated. After analysing the translation, we have discovered that further modifications of the translated dataset were required to generate a reliable Question Answering dataset. The translation result has been modified in two different ways: Manual and Automatic. These two methods generated two different versions of the ParSQuAD dataset.

Finally, both versions of the ParSQuAD dataset has been employed to train QA models, i.e., BERT (ParsBERT), ALBERT (ALBERT-Persian), and Multilingual-BERT (mBERT), which mBERT achieved scores of 56.66% and 52.86% for F1 score and exact match ratio respectively on the test set with the manual version and scores of 70.84% and 67.73% respectively with the automatic version.

In addition, the effect of the ratio of unanswerable questions in the dataset has been studied. After increasing the ratio of unanswerable questions in the first version of ParSQuAD, the scores improved by 3% on average.

Both versions of the ParSQuAD dataset are now publicly accessible for further usage in Open-domain Question Answering system implementations.

Although the models trained on our dataset have achieved acceptable scores and have been implemented and tested in a real-world QA system, our dataset may not have the quality of a native Persian Reading Comprehension dataset containing native question and answer samples annotated by multiple human annotators. Therefore, generating such datasets would be beneficial in future works.

ACKNOWLEDGMENT

This study is a work of the BIGDATA UI research group, and we would like to thank all the members.

REFERENCES

[1] F. Zhu, W. Lei, C. Wang, J. Zheng, S. Poria and T.-S. Chua, "Retrieving and Reading: A Comprehensive Survey on Open-domain Question Answering", *arXiv preprint arXiv:2101.00774*, 2021.

[2] D. Chen, A. Fisch, J. Weston and A. Bordes, "Reading Wikipedia to Answer Open-Domain Questions", *arXiv preprint arXiv:1704.00051*, 2017.

[3] S. Wang et al., "R3: Reinforced Ranker-Reader for Open-Domain Question Answering", in *Proceedings of the AAAI Conference on Artificial Intelligence*, 2018, vol. 32, no. 1.

[4] R. Das, S. Dhuliawala, M. Zaheer and A. McCallum, "Multi-step Retriever-Reader Interaction for Scalable Open-domain Question Answering", *arXiv preprint arXiv:1905.05733*, 2019.

[5] K. Guu, K. Lee, Z. Tung, P. Pasupat and M.-W. Chang, "Realm: Retrieval-Augmented Language Model Pre-Training", *arXiv preprint arXiv:2002.08909*, 2020.

[6] K. M. Hermann et al., "Teaching Machines to Read and Comprehend", in *Advances in neural information processing systems*, vol. 28, pp. 1693–1701, 2015.

[7] T. Nguyen et al., "MS MARCO: A Human Generated Machine Reading Comprehension Dataset", In *CoCo@ NIPS*, 2016.

[8] G. Lai, Q. Xie, H. Liu, Y. Yang and E. Hovy, "RACE: Large-scale Reading Comprehension Dataset from Examinations", *arXiv preprint arXiv:1704.04683*, 2017.

[9] P. Rajpurkar, J. Zhang, K. Lopyrev and P. Liang, "SQuAD: 100,000+ Questions for Machine Comprehension of Text", *arXiv:1606.05250 [cs]*, Oct. 2016, Accessed: Jan. 27, 2021. [Online]. Available: http://arxiv.org/abs/1606.05250

[10] P. Rajpurkar, R. Jia and P. Liang, "Know What You Don't Know: Unanswerable Questions for SQuAD", *arXiv:1806.03822 [cs]*, Jun. 2018, Accessed: Jan. 27, 2021. [Online]. Available: http://arxiv.org/abs/1806.03822

[11] Y. Wu et al., "Google's Neural Machine Translation System: Bridging the Gap between Human and Machine Translation", *arXiv preprint arXiv:1609.08144*, 2016.

[12] S. Lim, M. Kim and J. Lee, "KorQuAD1.0: Korean QA Dataset for Machine Reading Comprehension", *arXiv preprint arXiv:1909.07005*, 2019.

[13] P. Efimov, A. Chertok, L. Boytsov and P. Braslavski, "SberQuAD–Russian Reading Comprehension Dataset: Description and Analysis", in *International Conference of the Cross-Language Evaluation Forum for European Languages*, Cham: Springer, 2020, pp. 3–15.

[14] Y. Cui et al., "A Span-Extraction Dataset for Chinese Machine Reading Comprehension", *arXiv preprint arXiv:1810.07366*, 2018.

[15] M. d'Hoffschmidt, W. Belblidia, T. Brendlé, Q. Heinrich and M. Vidal, "FQuAD: French Question Answering Dataset", *arXiv:2002.06071 [cs]*, May 2020, Accessed: Oct. 05, 2020. [Online]. Available: http://arxiv.org/abs/2002.06071

[16] C. P. Carrino, M. R. Costa-jussà and J. A. R. Fonollosa, "Automatic Spanish Translation of the SQuAD Dataset for Multilingual Question Answering", *arXiv:1912.05200 [cs]*, Dec. 2019, Accessed: Oct. 05, 2020. [Online]. Available: http://arxiv.org/abs/1912.05200

[17] H. Mozannar, K. E. Hajal, E. Maamary and H. Hajj, "Neural Arabic Question Answering", *arXiv:1906.05394 [cs]*, Jun. 2019, Accessed: Oct. 05, 2020. [Online]. Available: http://arxiv.org/abs/1906.05394

[18] K. Lee, K. Yoon, S. Park and S. Hwang, "Semi-supervised Training Data Generation for Multilingual Question Answering", In *Proceedings of the Eleventh International Conference on Language Resources and Evaluation (LREC 2018)*, 2018.

[19] M. Artetxe, S. Ruder and D. Yogatama, "On the Cross-lingual Transferability of Monolingual Representations", *arXiv preprint arXiv:1910.11856*, 2019.

[20] P. Lewis, B. Oğuz, R. Rinott, S. Riedel and H. Schwenk, "MLQA: Evaluating Cross-Lingual Extractive Question Answering", *arXiv preprint arXiv:1910.07475*, 2019.

[21] N. Tohidi, C. Dadkhah and R. B. Rustamov, "Optimizing Persian Multi-Objective Question Answering System", *International Journal on Technical and Physical Problems of Engineering (IJTPE)*, vol. 13, no. 46, pp. 62–69, 2021.

[22] H. Veisi and H. F. Shandi, "A Persian Medical Question Answering System", *International Journal on Artificial Intelligence Tools*, vol. 29, no. 06, p. 2050019, 2020.

[23] Y. Boreshban, H. Yousefinasab and S. A. Mirroshandel, "Providing a religious corpus of question answering system in persian", *Signal and Data Processing*, vol. 15, no. 1, pp. 87–102, 2018.

[24] M. Joshi, E. Choi, D. S. Weld and L. Zettlemoyer, "Triviaqa: A large scale distantly supervised challenge dataset for reading comprehension", *arXiv preprint arXiv:1705.03551*, 2017.

[25] M. Farahani, M. Gharachorloo, M. Farahani and M. Manthouri, "ParsBERT: Transformer-based Model for Persian Language Understanding", *arXiv preprint arXiv:2005.12515*, 2020.

Negin Abadani received her B.Sc. of computer science from Univercity of Isfahan, in 2019. Presently, she is pursuing her M.Sc. in software engineering at University of Isfahan,. Her research intresets are in Question Answering, Natural Language Processing, Deep Learning and Data Mining. Currently, she is a member of the BIGDATA Research Group at University of Isfahan.

Jamshid Mozafari is a research assistant in Natural Language Processing and Information Retrieval at the BIGDATA lab of the University of Isfahan. He has received his B.Sc. and M.Sc. degrees in computer engineering from the University of Kurdistan and University of Isfahan in 2016 and 2019, respectively. His interests include Question Answering, Information Retrieval, and Machine Reading Comprehension. Currently, he is a member of the BIGDATA Lab at the University of Isfahan.

Afsaneh Fatemi received her B.S. degree in software engineering from Isfahan University of Technology in 1995, and the M.S. and Ph.D. degrees in software engineering both from University of Isfahan in 2002 and 2012, respectively. She is currently an assistant professor in the department of software engineering of University of Isfahan. Her current research interests include Big Data applications and challenges, especially in Question Answering Systems and social networks. She is also a member of Big Data Research Group of University of Isfahan from 2016.

Mohammadali Nematbakhsh is a full professor of software engineering in School of Computer engineering at the University of Isfahan. He received his B.Sc. in electrical engineering from Louisiana Tech University in 1981 and his M.Sc. and Ph.D. degrees in electrical and computer engineering from the University of Arizona in 1983 and 1987, respectively. He had worked for Micro Advanced Co. and Toshiba Corporation for many years before joining University of Isfahan. He has published more than 160 research papers, several US-registered patents and two database books that are widely used in universities. His main research interests include intelligent Web and big data processing. He is also the head of Big Data Research Group of University of Isfahan.

Arefeh Kazemi received her B.Sc. degree in software engineering and the M.Sc. degree in artificial intelligence from University of Isfahan, Isfahan, Iran, in 2008 and 2010, respectively. She obtained her Ph.D. degree in artificial intelligence field from University of Isfahan in 2017. Currently she is an assistant professor in Computational Linguistics branch in University of Isfahan. Her main areas of research interest include Natural Language Processing, Computational Linguistics and Data Mining.