Advances in Science, Technology and Engineering Systems Journal
Vol. 5, No. 5, 478-484 (2020)
www.astesj.com
Special Issue on Innovative Research in Applied Science, Engineering and Technology

ASTES Journal
ISSN: 2415-6698

Contextual Word Representation and Deep Neural Networks-based Method for Arabic Question Classification

Alami Hamza[*,1], Noureddine En-Nahnahi[1], Said El Alaoui Ouatik[1,2]

[1]*LISAC Laboratory, Faculty of Sciences Dhar El Mahraz, Sidi Mohamed Ben Abdellah University, PO Box 1796, Fez 30003, Morocco*

[2]*Laboratory of Engeneering Sciences, National School of Applied Sciences, Ibn Tofail University, Kenitra, Morocco*

ARTICLE INFO

Article history:
Received: 01 July, 2020
Accepted: 05 September, 2020
Online: 24 September, 2020

Keywords:
Arabic Question Classification
Contextual Word Representation
ELMo
RNN
CNN

ABSTRACT

Contextual continuous word representation showed promising performances in different natural language processing tasks. It stems from the fact that these word representations consider the context in which a word appears. But until recently, very little attention was paid to the contextual representations in Arabic question classification task. In the present study, we employed a contextual representation called Embeddings from Language Models (ELMo) to extract semantic and syntactic relations between words. Then, we build different deep neural models according to three types: Simple models, CNN and RNN mergers models, and Ensemble models. These models are trained on Arabic questions corpus to optimize the cross entropy loss given questions representations and their expected labels. The dataset consists of 3173 questions labeled according the Arabic taxonomy and an updated version of the Li & Roth taxonomy. We performed various comparisons with models based on the widely known context-free word2vec word representation. These evaluations confirm that ELMo representation achieves top performances. The best model scores up to 94.17%, 94.07%, 94.17% in accuracy, macro $F1$ *score, and weighted* $F1$ *score, respectively.*

1 Introduction

Question Answering Systems (QAS) have become one of the most popular information retrieval applications. These systems enable automatic answering to natural questions. This involves several parts functioning jointly to extract exact response, considering a user's question. A typical QAS is composed of three main components: 1) Question processing performs Question Classification (QC) and keywords extraction; 2) Passage retrieval apply information retrieval techniques to extract the passages that most probably contains the answer; 3) Answer processing processes the extracted passages and formulate the answer in natural language. A proper QC method enhances the performance of QAS by omitting insignificant answer candidates. In [1], the authors showed that 36.4% of errors made by QAS are associated to the QC. Despite the progress in different English natural language processing domains due to machine learning models, Arabic questions classification must address countless challenges owing to the lack of labeled corpora and difficulties associated to the complex morphology of this language, such us the absence of capital letters, the presence of diacritical marks, and its inflectional and derivational nature.

A typical QC method based on machine learning is composed of three main steps: 1) question preprocessing; 2) question representation; and 3) question classification. Previous works [2] represented words with term frequency-inverse document frequency (TF-IDF) method. This latter neglects relationships between words inside a question resulting in poor question representation. This restrictions led the authors [3] to use a continuous distributed word embeddings model [4]. This word representation ignores the context word yet it has the up side of considering both semantic and syntactic relations between words.

In this paper, we build question embedding by using a contextual representations namely Embeddings from Language Models (ELMo) [5]. Unlike the well known word2vec [4] representation that not consider the context, ELMo representation is able to compute word representation considering the word's context. In addition, we build various models according to three types: Simple models, CNN and RNN mergers models, and Ensemble models. The simple models are based on global max pooling, CNN, and RNN without any CNN/RNN combination. The CNN and RNN mergers models combine CNN and RNN layers to extract automatically more features from word representations. The Ensemble models predict

[*]Corresponding Author: Alami Hamza, Sidi Mohammed Ben Abdellah University, Morocco hamza.alami1@usmba.ac.ma

https://dx.doi.org/10.25046/aj050559

the probability of class labels by mixing the probability scores from different models. Our evaluations confirm that contextual representations show a good effects on Arabic question classification, and ensemble models based on word2vec and ELMo representations achieves top performances since they scored up to 94.17%, 94.07%, 94.17% in regards to accuracy, macro $F1$ score, and weighted $F1$ score.

The rest of the paper is structured as follows: Section 2 provides a brief overview of the previous works on Arabic question classification; Section 3 reveals our method for Arabic question classification which is based on contextual word representation; Section 4 presents the performance evaluations and the comparison results of various models; Finally, Section 5 concludes and gives future works and perspectives.

2 Related works

Numerous researches proposed methods for Arabic question classification. We introduce some of these methods in the following paragraphs.

In [6], the authors created a question answering system particularly for the holy Quran. Questions are represented by a set of terms, every term comprises both a part of speech tag and a stem of a word. They introduced a new taxonomy especially for classifying the question associated to the holy Quran. A support vector machine (SVM) classifier was trained on 180 training questions and tested with 3-folds cross validation on 50 questions. The obtained accuracy was about 77.2%.

In [7], the authors pursued a rule based approach to build their Arabic question classifier. The authors employed the NOOJ [1] tool to develop the Arabic linguistic rules that characterize question. They prepared a set of 200 questions for training questions and a set of 200 questions for test. They observed recall and precision are 93% and 100%, respectively.

In [2], the authors performed a comparison between two well known algorithms, SVM and Multinomial Naive Bayes, for Arabic question classification. In order to represent questions into vector space model, the TF-IDF method was applied. They trained the models with 300 training questions and tested on 200 questions. The best results were achieved by the SVM model which scored 100%, 94%, and 97% on precision, recall, and F1-measure.

In [3], the authors suggested a new Arabic taxonomy motivated from Arabic linguistic rules. They applied the continuous distributed word representation proposed in [4, 8, 9] to represent words. This representation captures semantic and syntactic relations between words. Various models including SVM, XGBoost, logistic regression were trained to classify questions given their words vectors. They built a dataset that contains 1041 questions for training and 261 questions for testing. The best performances were scored with the SVM model. It achieved 90%, 91%, 90%, and 90% on accuracy, precision, recall, and F1 score.

These works added on the progress of Arabic question classification methods. Nevertheless, the sizes of the datasets used in these studies were relatively small. Also, the TF-IDF representation has several drawbacks such us the word representation is sparse and huge, the semantic and syntactic relations between words can not be captured. What's more, the enriched word2vec representation calculates static word vectors neglecting the context were a word appears. Thus, a word with multiple meaning (polysemy phenomena) is misrepresented. Finally, the method based on linguistic rules proposed by [7] is highly impacted by the dataset used to extract rules and demands greater time and rules to consider additional Arabic question types.

3 Method

Our method is composed of three main steps including preprocessing, question representation, and questions classification.

3.1 Preprocessing

Our preprocessing pipeline is composed of two main steps: 1) punctuation and non Arabic words removal; 2) questions tokenization. Unlike standard pipelines of text preprocessing which perform stop words removal, we keep these words since they contain valuable information and are involved during the question classification. For instance, the words *Why* and *Who* are essential to identify the question type.

3.2 Question representation

To represent questions into a vector space format, we use the ELMo representation introduced in [5, 10]. All the words included into a question are passed to a neural network that is composed of two main layers: 1) one dimensional convolutional neural network with different filter sizes that computes word embeddings based on its character level embeddings; 2) a stack of two Bidirectional LSTM (BiLSTM) layers. The network is trained with large textual corpora to optimize language model objective. Thus, the ELMo representation of a word k, given by $\mathbf{ELMo}_k$, is calculated by the next equation:

$$\mathbf{ELMo}_k = \frac{1}{3}\sum_{j=0}^{2}\mathbf{h}_{k,j}^{(LM)} \tag{1}$$

where $\mathbf{h}_{k,j}$ is the output of the hidden layers j of the neural network. Figure 1 depicts the overall design of the ELMo technique. In order to calculate contextual word representation, we used the pre-trained word representation presented by [10, 11]. The model is trained with a set of 20-million-words data randomly picked from the Arabic Wikipedia corpus. Every word in represented by an 1024 dimension vector.

A question Q, that is a succession of l words, is modeled by the later matrix:

$$\mathbf{Q} = [\mathbf{ELMo}_{word_1}, ..., \mathbf{ELMo}_{word_l}] \tag{2}$$

[1] http://www.nooj-association.org/

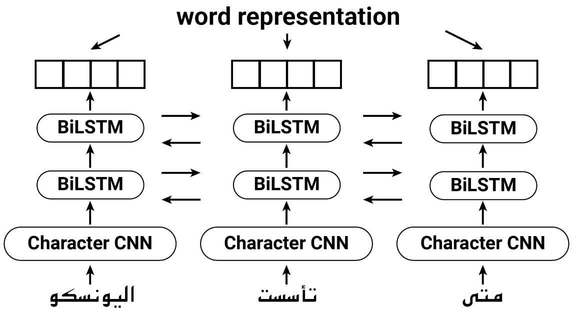

Figure 1: The overall architecture of Embeddings from Language Models

3.3 Questions classification

We constructed several neural network models on top of ELMo embeddings for Arabic questions classification. The goal is to investigate the behavior of contextual representation with diverse neural network architectures. In the course of training, the parameters of the ELMo model are fixed and the remainder of the parameters are optimized according the categorical cross entropy loss denoted by the equation:

$$J(\theta) = -\frac{1}{T}\sum_{t=1}^{T}\left[y_t \log \hat{y}_t + (1-y_t)\log(1-\hat{y}_t)\right] \qquad (3)$$

where y_t is the true class label, $\hat{y}_t$ is the predicted class label, and T is the count of a taxonomy's classes.

3.3.1 ELMo with global max pooling

Our primary model consists of a pile of layers including input layer, global max pooling layer and a softmax layer. The input layer represents the input features which are questions embeddings calculated with the ELMo model. The global max pooling catches the more important characteristics. The softmax layer maps these features to a vector of probabilities. Figure 2 depicts the explained architecture.

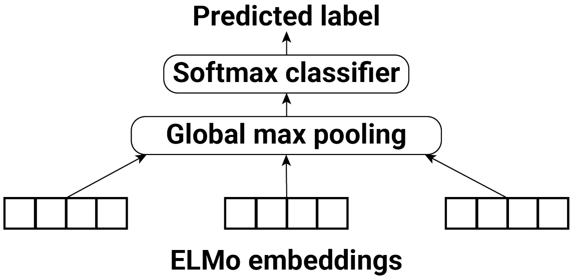

Figure 2: ELMo with global max pooling model

3.3.2 ELMo with global max pooling and dense layer

Based on the model described in the previous subsection 3.3.1 and with the purpose of extracting more features, we included a dense layer among the softmax layer and the global max pooling layer. Figure 3 illustrates the described architecture.

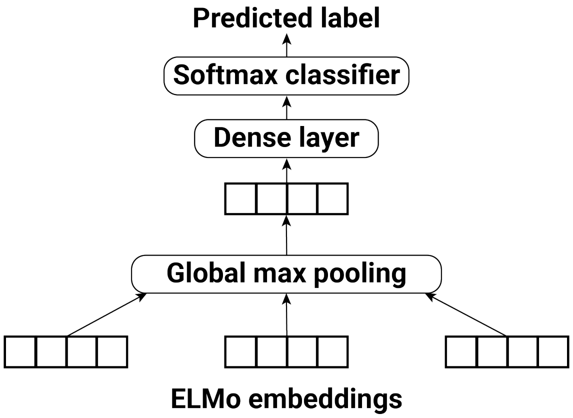

Figure 3: ELMo with global max pooling and dense layer model

3.3.3 ELMo with last hidden state of a GRU layer

To extract additional features, we included a Gated Recurrent Unit (GRU) layer that can handle sequential data and extract helpful information. Let denote Q a question represented by $Q = [w_1, w_2, ...w_k]$ where k the number of word vectors within Q and w_i is the i-th word vector of the question. GRU processes the data word-by-word according to the time-step from the past to the future. At each time step the current hidden state is computed as follows:

$$r_t = \text{sigmoid}\left(W_r \cdot [h_{t-1}, w_t]\right) \qquad (4)$$

$$z_t = \text{sigmoid}\left(W_z \cdot [h_{t-1}, w_t]\right) \qquad (5)$$

$$\tilde{h}_t = \tanh\left(W_{\tilde{h}} \cdot [r_t \odot h_{t-1}, w_t]\right) \qquad (6)$$

$$h_t = (1-z_t) \odot h_{t-1} + z_t \odot \widetilde{h_t} \qquad (7)$$

The output of the final hidden state is passed to the softmax layer to compute the probabilities of belonging to each class label. Figure 4 presents the described architecture.

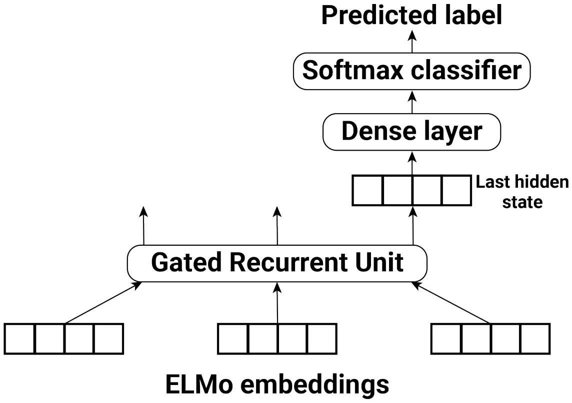

Figure 4: ELMo with last hidden state of a GRU layer model

3.3.4 ELMo with GRU and global max pooling

This model considers all the time steps from the GRU layer. It applies a global max pooling to extract the most important features from every words within a question. The architecture of the model is illustrated in Figure 5.

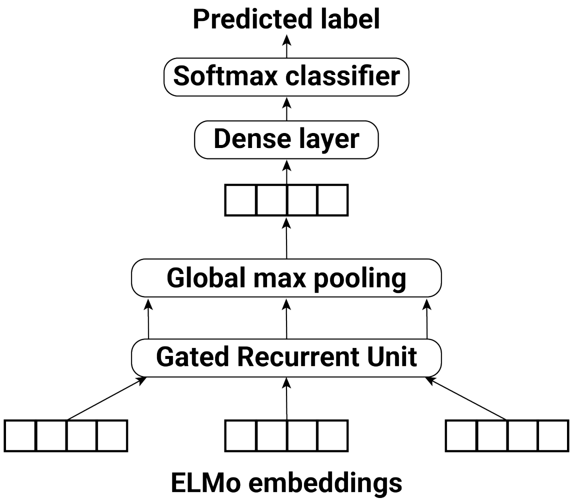

Figure 5: ELMo with GRU and global max pooling model

3.3.5 ELMo with CNN and global max pooling

Convolutional Neural Networks (CNN) proved good performances in both computer vision and NLP fields. These models extract new features applying 1 dimensional convolution on word neighbours. Figure 6 illustrates the model architecture. Let $w_i \in \mathbb{R}^p$ the p dimensional word representation corresponding to the i-th word in the question. A question of length k, which is padded when necessary, is represented by the following equation:

$$Q = [w_1, w_2, ...w_k] \tag{8}$$

where Q the question matrix. New features $m \in \mathbb{R}^{k-n+1}$ are produced by applying a convolution to each window of n words. The window size n vary between 2 to 5 thus we compute for each widow size a feature map. We then apply global max pooling [12] to capture the most important features. We concatenate these features and apply a softmax layer to classify the question.

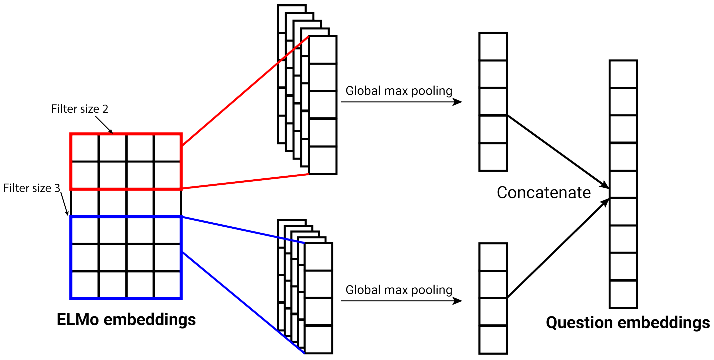

Figure 6: Multiple filters CNN with global max pooling

3.3.6 ELMo with CNN, GRU, and global max pooling

This model apply a convolution filters on top of ELMo embeddings. This operation results in a number of features maps equals to the number of filters used. Next, each features map is passed to GRU layers. The question embeddings is then the concatenation of the outputs of a set global max pooling functions applied on top of each hidden states of GRU layers. Finally, a softmax function is applied to classify the question. Figure 7 illustrates the architecture of this model.

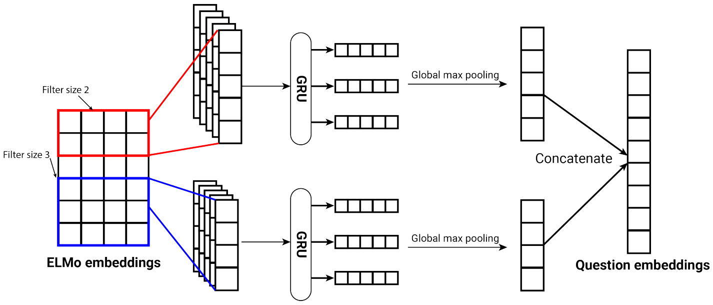

Figure 7: Multi filters CNN and GRU layers

3.3.7 ELMo with GRU, CNN, and global max pooling

The architecture of this model is similar to the architecture described in section 3.3.6. The only change is we apply first a GRU layer then convolution operations with different filter. Figure 8 presents the general architecture of this model. Various convultions with filter size between 2 to 5 are applied to the hidden states of the GRU layer. Next, a global max pooling is applied to extract the most important features for each filter features. Finally, the outputs of the global max pooling functions are concatenated to build the question embeddings.

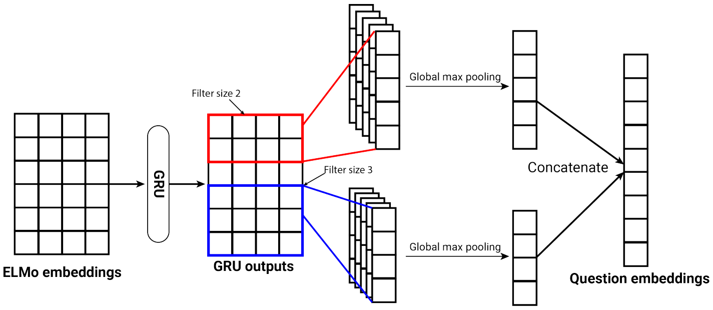

Figure 8: GRU and multi filters CNN model

3.3.8 Ensemble Models

An ensemble model aggregates the prediction of diverse models results in once final prediction for unseen data. We applied two ensemble models including mean scores-based ensemble and max scores-based ensemble. The mean scores-based model computes the average of the probability scores predicted by each model the class label with the highest probability score is assigned to the question. The max scores-based model pick the class label with the highest probability across all the predicted scores from different model.

4 Experimental results

4.1 Questions Classification Dataset

The dataset consists of 3173 Arabic questions. These questions are manually labeled with an application created by our team. We picked randomly 80% (2538) of questions as training set and 20% (635) of questions as testing set. Table 1 and Table 2 show the distribution of the class labels.

Table 1: Distribution of classes in the Arabic questions dataset

Classes	#	Classes	#
(Human...) العاقل	602	(Entity...) غير العاقل	1084
(Status...) حال الشيء و هيئته	121	(Location) المكان	450
(Time) الزمان	318	(Numbers) العدد	304
(Yes/No) التصديق	294		

Table 2: Distribution of classes in the Arabic questions dataset

Classes	#	Classes	#
Abbreviation	22	Description	709
Entity	468	Human	627
Location	453	Numeric	600
Yes/No	294		

4.2 Experimental settings

Considering that we deal with a multi-class classification problem, we use three measures to evaluate every model. The measures contain accuracy, macro F1 score and weighted F1 score. We employed widely known libraries including Scikit-learn, Tensorflow 2.0 and Keras libraries to construct and evaluate every single model [13, 14]. In the interest of reducing the internal covariate shift and the training time, we utilize layer normalization [15]. We fixed the batch size at 32 questions and the epochs at 1000 iterations for each model. We retained the model that achieves the optimal loss on training set. Our experiments are completely carried out in Google Colaboratory[2].

4.3 Simple models evaluation

We segment the proposed models into three sets including simple models, CNN and RNN mergers, ensemble models. The first set contains simple models that are composed with global max pooling, convolutions, and recurrent neurones layers. These model do not contain any type of combination between convolutions and recurrent neurones layers. The second set is named CNN and RNN mergers where convolutions and recurrent features are merged. The last models apply ensemble methods to build the question classification performance.

We evaluate our simple models with two different word representation: 1) The context-free word representation, designated by enriched word2vec with subword information [16], which embeds a word into a 300 dimensional vector. This technique possesses the capability to catch semantic and syntactic relationships among words. At the same time, it do not take into account the word in context; and 2) The contextual word representation ELMo [5] which calculates a 1024 dimensional vector for each word. This representation regards the context of word resulting in an enhanced question embeddings. Table 3 and Table 4 shorten the achieved performances with both the Arabic taxonomy and the updated Li & Roth taxonomy. In the event of the Arabic taxonomy, the ELMo representation surpasses the enriched word2vec representation in regards to accuracy, macro F1 score, and weighted F1 score. Nonetheless, this come at the expense of the lower size of the words' vectors. Lastly, the model which consists of ELMo embeddings, GRU layer, and global max pooling layer achieves the best performances outlined by 93.86%, 93.37%, and 93.84% on accuracy, macro F1 score, and weighted F1 score. In the event of the updated Li & Roth taxonomy, word2vec based models handle better the imbalanced dataset problem. The model that have word2vec embeddings as inputs and composed of GRU and global max pooling achieves 87.24% in terms of macro F1 score. The model composed of ELMo embeddings, CNN, and global max pooling scored top results in terms of accuracy and weighted F1 score.

Table 3: Performance measures (accuracy, macro F1 score, and weighted F1 score) of simple models with Arabic taxonomy

	Accuracy	*macro F1 score*	*weighted F1 score*
W2V + Global max pooling	78.74%	78.38%	78.64%
W2V + Global max pooling + dense	82.99%	82.86%	82.92%
W2V + Last hidden state of GRU	92.44%	92.74%	92.45%
W2V + GRU + Global max pooling	92.76%	92.96%	92.75%
W2V + CNN + Global max pooling	90.55%	91.00%	90.56%
ELMo + Global max pooling	86.77%	88.25%	86.83%
ELMo + Global max pooling + dense	88.03%	89.03%	88.11%
ELMo + Last hidden state of GRU	91.97%	92.12%	91.96%
ELMo + GRU + Global max pooling	**93.86%**	**93.37%**	**93.84%**
ELMo + CNN + Global max pooling	90.70%	91.12%	90.65%

Table 4: Performance measures (accuracy, macro F1 score, and weighted F1 score) of simple models with the updated Li & Roth taxonomy

	Accuracy	*Macro F1 score*	*Weighted F1 score*
W2V + Global max pooling	76.54%	65.61%	76.32%
W2V + Global max pooling + dense	81.73%	70.51%	81.68%
W2V + Last hidden state of GRU	91.65%	85.50%	91.56%
W2V + GRU + Global max pooling	92.13%	**87.24%**	92.05%
W2V + CNN + Global max pooling	89.13%	85.87%	88.97%
ELMo + Global max pooling	80.94%	71.87%	80.52%
ELMo + Global max pooling + dense	84.72%	76.51%	84.86%
ELMo + Last hidden state of GRU	91.50%	82.31%	91.69%
ELMo + GRU + Global max pooling	91.97%	82.67%	92.03%
ELMo + CNN + Global max pooling	**92.28%**	82.78%	**92.33%**

4.4 CNN and RNN mergers evaluation

We test out CNN and RNN mergers with the word2vec and ELMo word representations. These models have the ability to extract more valuable features from raw data. Table 5 presents the results of the CNN and RNN mergers with the Arabic taxonomy. The model that takes ELMo embeddings as inputs and apply GRU then CNN achieves the best scores 94.01%, 93.60%, and 93.98% in terms of accuracy, macro F1 score, and weighted F1 score. Besides, Table 6 shows that the word2vec word representation with Li & Roth taxonomy scored the best results 91.81% accuracy, 88.32% macro F1 score, and 91.84% weighted F1 score. However, the model

[2] https://colab.research.google.com/

architecture remains the same. Thus, the evaluation confirms that applying GRU layer followed by CNN layer is better than applying CNN layer then GRU layer for Arabic question classification.

Table 5: Performance measures (accuracy, macro F1 score, and weighted F1 score) of the CNN and RNN mergers with the Arabic taxonomy

	Accuracy	*Macro F1 score*	*Weighted F1 score*
W2V+CNN+GRU+Global Max Pooling	90.71%	90.91%	90.68%
W2V+GRU+CNN+Global Max Pooling	92.44%	92.47%	92.43%
ELMo+CNN+GRU+Global Max Pooling	91.02%	90.96%	91.03%
ELMo+GRU+CNN+Global Max Pooling	**94.01%**	**93.60%**	**93.98%**

Table 6: Performance measures (accuracy, macro F1 score, and weighted F1 score) of the CNN and RNN mergers with the updated Li & Roth taxonomy

	Accuracy	*Macro F1 score*	*Weighted F1 score*
W2V+CNN+GRU+Global Max Pooling	91.02%	86.16%	90.92%
W2V+GRU+CNN+Global Max Pooling	**91.81%**	**88.32%**	**91.84%**
ELMo+CNN+GRU+Global Max Pooling	90.87%	81.24%	90.86%
ELMo+GRU+CNN+Global Max Pooling	91.65%	84.87%	91.75%

4.5 *Ensemble models evaluation*

Finally, we evaluate the ensemble models based on the mean scores and the max scores obtained by classifier models with the best performances. Table 7 and Table 8 present the result of the ensemble model based on the GRU + CNN + Global Max Pooling model with Arabic and Li & Roth taxonomies, respectively. We notice that the ensemble method surpasses simple models and CNN and RNN mergers in the case of Arabic taxonomy. However, for Li & Roth taxonomy the mean scores-based ensemble model achieved the best results in terms of accuracy and weighted F1 score. The GRU + Global Max Pooling achieves the best performance according to the macro F1 score.

Table 7: GRU + CNN + Global Max Pooling ensemble model with Arabic taxonomy

	Accuracy	Macro F1 score	Weighted F1 score
Mean scores-based Ensemble	94.17%	94.07%	94.17%
Max scores-based Ensemble	94.17%	94.07%	94.17%

Table 8: GRU + CNN + Global Max Pooling ensemble model with Li & Roth taxonomy

	Accuracy	Macro F1 score	Weighted F1 score
Mean scores-based Ensemble	**92.60%**	**86.52%**	**92.69%**
Max scores-based Ensemble	92.44%	86.39%	92.54%

4.6 *Discussion*

From the one hand we discuss the obtained results for the Arabic taxonomy. The ELMo contextual word representation performs better than the word2vec context free word representation. The simple models evaluation shows that the GRU a recurrent neural network architecture is more appropriate than CNN for Arabic question classification. The CNN and RNN mergers evaluation confirms that the stack GRU/CNN achieves better results than the stack CNN/RNN. Besides, the ensemble models build with the word2vec and ELMo embeddings impact positively the performance of the Arabic question classification task. On the other hand, the classifiers trained with Li & Roth taxonomy shows different behaviors. The simple models evaluation reveals that the ELMo representation with CNN + Global Max Pooling model has top performances in terms of accuracy and weighted F1 score while the word2vec representation with GRU + Global Max Pooling model has the top performance in terms of macro F1 score. Thus, the latter model is able to handle imbalanced data problem more appropriately. The CNN and RNN mergers evaluation supports that word2vec with the stack GRU/RNN model is better than ELMo embeddings with RNN/GRU model. Finally, ensemble models improve the accuracy and the weighted F1 score but not the macro F1 score.

To conclude, the Arabic taxonomy works well with contextual representation ELMo and ensemble models while the model choice with Li & Roth taxonomy is related to the performance, e.g., if the objective is to optimize the accuracy of the question classification task then the ensemble model is more appropriate.

5 Conclusion

In this work, we built various Arabic question classifier based on simple models, CNN and RNN mergers and Ensemble methods. We trained these models with both context free word representation word2vec [4, 16] and contextual word representation ELMo [5, 10]. The latter has the upside to compute, for a word, a vector that catch semantic and syntactic meaning by considering its context. Ability that the enriched wor2vec model can not perform. We compared the performances of the proposed models with two different question taxonomies including Arabic taxonomy and Li & Roth taxonomy. From the one side, the experiments on questions labeled with Arabic taxonomy showed that contextual representation achieved promising results 94.01% accuracy, 93.60% macro F1 score and 93.98% weighted F1 score. Along with, ensemble methods improve the results slightly since it scored the top performances 94.17% accuracy, 94.07% macro F1 score and 94.17% weighted F1 score. On the other side, for questions labeled with Li & Roth taxonomy the top classifier in terms of macro F1 score (87.24%) was built based on word2vec and GRU + Global Max Pooling model. The mean score-based ensemble model scored 92.60% accuracy and 92.69% weighted F1 score which are the best obtained results in terms of accuracy and weighted F1 score. Thus, the the model choice with Li & Roth taxonomy is related to the performance the classifier needs to optimize more

As perspectives, we arrange to expand this work by building other Arabic question answering system components including the passage retriever and the answer processing modules. First, we plan to construct a module that aims to retrieve the most similar passages to a question based on contextual representation. Next, We intend to integrate these context aware representations in the answer processing module. Finally, we project to build an Arabic question answering system based on components that integrate contextual word embeddings, which have the capabilities of extracting syntactic and semantic relation of a word considering its context, to enhance further their performances.

Conflict of Interest The authors declare no conflict of interest.

References

[1] D. Moldovan, M. Paşca, S. Harabagiu, M. Surdeanu, "Performance issues and error analysis in an open-domain question answering system," ACM Transactions on Information Systems, **21**(2), 133–154, 2003, doi:10.1145/763693.763694.

[2] W. Ahmed, B. A. P, "CLASSIFICATION OF ARABIC QUESTIONS USING MULTINOMIAL NAIVE BAYES." International Journal of Latest Trends in Engineering and Technology Special Issue SACAIM, 82–86, 2016.

[3] A. Hamza, N. En-Nahnahi, K. A. Zidani, S. E. A. Ouatik, "An arabic question classification method based on new taxonomy and continuous distributed representation of words," Journal of King Saud University - Computer and Information Sciences, 2019, doi:https://doi.org/10.1016/j.jksuci.2019.01.001.

[4] P. Bojanowski, E. Grave, A. Joulin, T. Mikolov, "Enriching Word Vectors with Subword Information," Trans. Assoc. Comput. Linguistics, **5**, 135–146, 2017.

[5] M. E. Peters, M. Neumann, M. Iyyer, M. Gardner, C. Clark, K. Lee, L. Zettlemoyer, "Deep Contextualized Word Representations," in M. A. Walker, H. Ji, A. Stent, editors, Proceedings of the 2018 Conference of the North American Chapter of the Association for Computational Linguistics: Human Language Technologies, NAACL-HLT 2018, New Orleans, Louisiana, USA, June 1-6, 2018, 1 (Long Papers), 2227–2237, Association for Computational Linguistics, 2018, doi:10.18653/v1/n18-1202.

[6] H. Abdelnasser, M. Ragab, R. Mohamed, A. Mohamed, B. Farouk, N. El-Makky, M. Torki, "Al-Bayan: an arabic question answering system for the holy quran," in Proceedings of the EMNLP 2014 Workshop on Arabic Natural Language Processing (ANLP), 57–64, 2014.

[7] H. M. Al Chalabi, S. K. Ray, K. Shaalan, "Question classification for Arabic question answering systems," in Information and Communication Technology Research (ICTRC), 2015 International Conference on, 310–313, IEEE, 2015.

[8] T. Mikolov, I. Sutskever, K. Chen, G. S. Corrado, J. Dean, "Distributed Representations of Words and Phrases and their Compositionality," in C. J. C. Burges, L. Bottou, Z. Ghahramani, K. Q. Weinberger, editors, Advances in Neural Information Processing Systems 26: 27th Annual Conference on Neural Information Processing Systems 2013. Proceedings of a meeting held December 5-8, 2013, Lake Tahoe, Nevada, United States, 3111–3119, 2013.

[9] T. Mikolov, K. Chen, G. Corrado, J. Dean, "Efficient Estimation of Word Representations in Vector Space," in Y. Bengio, Y. LeCun, editors, 1st International Conference on Learning Representations, ICLR 2013, Scottsdale, Arizona, USA, May 2-4, 2013, Workshop Track Proceedings, 2013.

[10] W. Che, Y. Liu, Y. Wang, B. Zheng, T. Liu, "Towards Better UD Parsing: Deep Contextualized Word Embeddings, Ensemble, and Treebank Concatenation," in D. Zeman, J. Hajic, editors, Proceedings of the CoNLL 2018 Shared Task: Multilingual Parsing from Raw Text to Universal Dependencies, Brussels, Belgium, October 31 - November 1, 2018, 55–64, Association for Computational Linguistics, 2018, doi:10.18653/v1/k18-2005.

[11] A. Kutuzov, M. Fares, S. Oepen, E. Velldal, "Word vectors, reuse, and replicability: Towards a community repository of large-text resources," in Proceedings of the 58th Conference on Simulation and Modelling, 271–276, Linköping University Electronic Press, 2017.

[12] R. Collobert, J. Weston, L. Bottou, M. Karlen, K. Kavukcuoglu, P. P. Kuksa, "Natural Language Processing (Almost) from Scratch," J. Mach. Learn. Res., **12**, 2493–2537, 2011.

[13] M. Abadi, A. Agarwal, P. Barham, E. Brevdo, Z. Chen, C. Citro, G. S. Corrado, A. Davis, J. Dean, M. Devin, S. Ghemawat, I. Goodfellow, A. Harp, G. Irving, M. Isard, Y. Jia, R. Jozefowicz, L. Kaiser, M. Kudlur, J. Levenberg, D. Mané, R. Monga, S. Moore, D. Murray, C. Olah, M. Schuster, J. Shlens, B. Steiner, I. Sutskever, K. Talwar, P. Tucker, V. Vanhoucke, V. Vasudevan, F. Viégas, O. Vinyals, P. Warden, M. Wattenberg, M. Wicke, Y. Yu, X. Zheng, "TensorFlow: Large-Scale Machine Learning on Heterogeneous Systems," 2015, software available from tensorflow.org.

[14] M. Abadi, P. Barham, J. Chen, Z. Chen, A. Davis, J. Dean, M. Devin, S. Ghemawat, G. Irving, M. Isard, M. Kudlur, J. Levenberg, R. Monga, S. Moore, D. G. Murray, B. Steiner, P. Tucker, V. Vasudevan, P. Warden, M. Wicke, Y. Yu, X. Zheng, "TensorFlow: A system for large-scale machine learning," in Proceedings of the 12th USENIX Symposium on Operating Systems Design and Implementation (OSDI '16), 265–283, 2016.

[15] L. J. Ba, J. R. Kiros, G. E. Hinton, "Layer Normalization," CoRR, **abs/1607.06450**, 2016.

[16] E. Grave, P. Bojanowski, P. Gupta, A. Joulin, T. Mikolov, "Learning Word Vectors for 157 Languages," in N. Calzolari, K. Choukri, C. Cieri, T. Declerck, S. Goggi, K. Hasida, H. Isahara, B. Maegaard, J. Mariani, H. Mazo, A. Moreno, J. Odijk, S. Piperidis, T. Tokunaga, editors, Proceedings of the Eleventh International Conference on Language Resources and Evaluation, LREC 2018, Miyazaki, Japan, May 7-12, 2018, European Language Resources Association (ELRA), 2018.