

A Neural Question Answering System for Supporting Software Engineering Students

Marco Antônio Calijorne Soares
LAIS
FUMEC University
Belo Horizonte, Brazil
marco.calijorne@techsysinfo.com

Wladimir Cardoso Brandão
IRIS
PUC Minas
Belo Horizonte, Brazil
wladmir@pucminas.br

Fernando Silva Parreiras
LAIS
FUMEC University
Belo Horizonte, Brazil
fernando.parreiras@fumec.br

Abstract—QA (Question Answering) is the task of automatically answer natural language questions posed by humans. Usually, QA approaches use a combination of computational linguistics, information retrieval and knowledge representation to find answers for questions. In a teaching-learning process, it is critical that teachers use a range of teaching strategies to effectively meet the needs of individual learners. Thus, QA approaches can be effectively used to support the teaching-learning process. In this article, we exploit neural networks for QA to support the teaching-learning process. Particularly, we use DMN+ (improved dynamic memory networks) and SeqToSeq (sequence to sequence) with a corpus of SE (software engineering) texts to effectively answer questions commonly posed by SE learners. Experimental results show that DMN+ is more effective than SeqToSeq for this task with up to 77% accuracy.

Index Terms—question answering, improved dynamic memory network, sequence to sequence, software engineering

I. INTRODUCTION

QA (Question Answering) has been an important subject of research and a widely investigated problem in the AI (Artificial Intelligence) field [1]. QA approaches retrieve relevant answers to natural language questions posed by users. In other words, a user can ask a question in natural language to a computational system, which will formulate an unique answer using a previously processed data collection, and the answer will be delivered to the user also in natural language. Asking questions in natural language is the most spontaneous way to manifest an information need [2], and, in this context, IR (information retrieval) [3], [4] and NLP (natural language processing), and KB (knowledge base) [5] techniques are particularly useful to address this class of problems.

Particularly, the educational environment can benefit greatly from QA approaches. Their performance was presented in several investigations reported on literature, but there is a lack of works that address the Portuguese language. Thus, we raised a research question: what is the performance of QA approaches when applied to a teaching-learning environment for a subject represented in Portuguese language?

In this article, we propose a neural QA system to support SE (software engineering) students in their learning process. For this, we implement and compare two QA techniques based on neural networks in order to understand how would be

their performance when applied to the SE domain using a Portuguese dataset. In particular,

Experiments attest the effectiveness of our neural QA system. Experimental results show that our approach achieve up to 77.00% of accuracy using DMN+. The major contributions of this article are the following: i) We propose a novel neural QA system to support SE students in their learning process; ii) We thoroughly evaluate our proposed system, showing that DMN+ performs better than SeqToSeq in the context of QA for Portuguese language.

The remainder of this article is organized as follows: In Section II, we review the related literature on question answering systems and neural networks, particularly on the DNM+ (improved dynamic memory network) and SeqToSeq (Sequence to sequence) algorithms used in our proposed system. In Section III, we describe our experimental procedures. In Sections IV and V, we present the results of the experimental evaluation of our approach, including the datasets used to attest its effectiveness. Finally, in Section VI, we present our concluding remarks, as well as directions for future research.

II. BACKGROUND

QA systems aim to answer questions asked in natural language using either a pre-structured database or a collection of written information [3], [4]. They are an advanced form of IR [6] systems. There are articles reported in literature that present the basic architecture of the QA systems [7]–[9], dividing them into three components: *question processing*, *document processing*, and *answer processing*.

Particularly, the *question processing* component is responsible to analyze the structure of a question and classify its morphology [7], [10]. Besides that, it classifies the type of the question [10], [11], and performs a question transformation to create a meaningful question formula compatible with QA's domain [12]. We can classify the question as *factoid*, which are the questions that are arguing about a fact and their answers do not use a lot of words [13], *definition*, which are the questions that require a summary or short passages as a answer [13], and *list*, which are questions that demands for their answers a set of entities that satisfy a given criteria [9]. The *document processing* component is responsible for written information understanding throw machine learning

and deep learning techniques [7], [9]. The *answer processing* component is responsible to execute extraction techniques on corpus information [8].

A number of techniques have been developed to implement question answering systems, more recently neural networks [14]. As reported in the literature, different types of neural networks are being used as the main technique to develop QA system [15], [16], and they are reaching outstanding results. Besides that, these techniques can be easily adapted to be used in a Portuguese corpus, which is one of our requirements in this article. In particular, we used two neural network algorithms to perform our experiments: the DMN+ (improved dynamic neural network), and SeqToSeq (sequence to sequence) based on RNN (recurrent neural network). We choose the DMN+ due its main nature, since it is an optimization of neural network architecture built for QA task. In addition, we choose SeqToSeq since it is being used in dialog and QA tasks as well [17].

A. Improved Dynamic Memory Network

An Improved Dynamic Memory Network (DMN+) is an enhancement of a Dynamic Memory Network (DMN) [16], a special type of neural network improved for addressing QA problems. From a training set of input sequences, that could be a sentence, a story, papers, books and questions, DMN can create episodic memories, using them to postulate consistent answers. The basic architecture of DMN for QA is composed by four components: *input*, *question*, *episodic memory*, and *answer* [18]. Figure 1 shows how these components interact among each other.

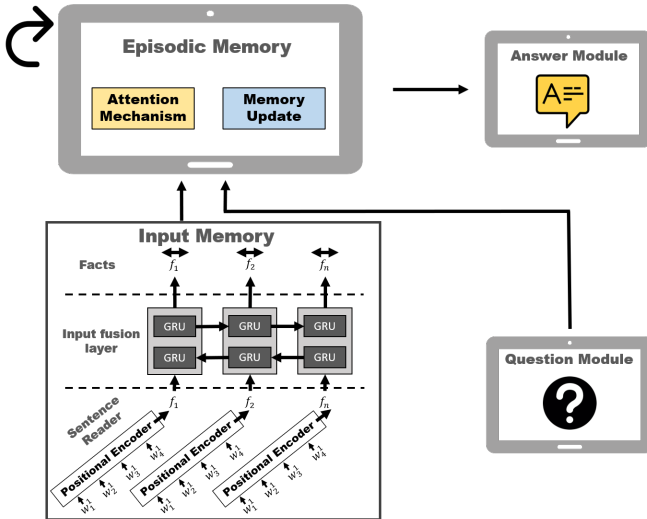


Fig. 1. Architecture of a QA system based on DMN+. The *input* component with the *sentence reader* and *input fusion layer* differentiates DMN+ from traditional DMN.

The *input* component is responsible to process the training data. It processes the input vectors associated with a question into a set of fact vectors. The component is built using a

GRU (Gated Recurrent Unit) that enables the network to learn either if the sentence being considered is relevant or it is not related to the answer. The *question* component processes each question word by word and creates a vector using the same GRU with the same weights as the *input* component. At this moment, both facts and questions are encoded as embedding. The *episodic memory* component is responsible to retrieve the answer for the question from the input facts. This component presents two sub-components, the attention mechanism responsible to create a contextual vector, and the memory update mechanism, that generates the episode memory based on the contextual vector. The *answer* component is responsible to generate an appropriate response.

The DMN+ addresses two main problems of DMN. The first is related to the single GRU problem, i.e., single GRU only allows a sentence to have context from sentences before it. The second is related to distance, i.e., if the related sentence (which could be the answer for example) is too far away, influencing on the interaction of these distance sentences on the word level GRU. These two problems were addressed by replacing the single GRU by two components, a *sentence reader* component and an *input fusion layer*, making possible the interactions between sentences [16]. From Figure 1, we observe the architecture of a DMN+.

B. Sequence To Sequence

Sequence to sequence (SeqToSeq) is a model based on two RNNs, one is the encoder and the other is the decoder. The encoder acts processing the input sequence, returning its own internal state. The outputs are discarded from the encoder RNN, only recovering the state. This state is used as the context of the decoder, trained to predict the next characters of the target sequence, given previous characters of the target sequence. Particularly, it is trained to turn the target sequences into the same sequences but offset by one timestep in the future, a training process called “teacher forcing” in this context. Note that, the encoder uses as initial state the state vectors from the encoder, which is how the decoder obtains information about what it is supposed to be generated [17], [19], [20].

To use a SeqToSeq model as a basis for a QA system, it must be trained as follows. The RNN encoder process the story (which is small segments of the corpus), followed by a symbol that determines the beginning of a question. Then, another symbol indicates to the network to starts decoding, with the decoder’s initial state being the encoder’s final. The decoder creates an answer sequence, followed by a STOP symbol which indicates when the processing should end.

III. EXPERIMENT

To validate our QA system, we exploit it on a teaching-learning process. In particular, we exploit it considering two distinct dimensions:

- *Domain of Application:* We defined a restrict domain subject where a QA system could be analyzed. We choose the SE subject, a computer science topic. We made this

choice due to its theoretical nature, which fits better to this kind of systems.

- *Neural Networks*: We implemented two different neural network approaches, DMN+ and SeqToSeq, to assess the QA system performance.

A. QA Domain and Corpus

An essential part of QA systems is the corpus used as its knowledge source for training and testing. In this article, we used the restrict domain of software engineering. The scientific literature has revealed that restricted domains QA systems provide accurate answers than open domains QA systems [4], [21], [22]. We choose SE subject due to its theoretical nature, which make the QA system task easier when answering fact, definition or list questions [4], [21], [23].

The experiments were carried out by using a corpus extracted from 11 SE books described in Table I. We choose these books because:

- They are the most popular books used by Brazilian professors in their SE lectures.
- Besides concepts, methods and standards of SE, some of these books also describe agile methods, a recurrent and important topic on this area.

TABLE I
SE BOOKS USED TO CREATE THE CORPUS

Book main Subject	Books
Software Engineering	[24], [25], [25]–[29]
Agile Methods	[30]–[33]

Additionally to the book content, we include 4,186 questions and answers pairs extracted from internet. The set composed by the books and question answers pairs were used as the training set. Moreover, 90 questions were prepared to be used as the testing set. This set was extracted from review questions chapters of the books.

B. Training and Testing sets

As described in section III-A, we used as a source of knowledge 11 SE books written in Portuguese. To extract the information of each book we defined a process and collected the data using a custom software written in python. After the books files normalization, we used the custom software to recover as much information as possible from files. To maximize the performance of this task, we established a set of exclusion criteria designed to guide the software on the content classification, making possible to remove non-important information. Table II provides the criteria we used to exclude content.

The software was designed to extract the information in phrases and save them as a new line in a text file. This made easier our task to deploy a corpus in the specific format requested by each QA algorithm we used. To improve the corpus quality we design a training questions set. As we need reliable questions and answers pairs, the data were gathered from a

TABLE II
EXCLUSION CRITERIA USED TO TO EXCLUDE CONTENT

Type of exclusion	Description
Titles, Headers and Footers	Books names, chapters description, sessions titles, headers and footers information were removed.
Sentences with less than 5 Words	We did not remove all sentences that met this criteria. Phrases that were part of a list such as bullets and numbering and sentences that were a part of the text were kept on the corpus.
Chapter or Section Description	Phrases that describes what each chapter or section talks about were removed.
Titles, Headers and Footers	Books names, chapters description, sessions titles, headers and footers information were removed.
Summaries and References	All summaries, tables of contents, figure lists and the references were removed.
Others	Page numbers, Proper names (Authors and Co-authors), References and specific phrases (manually identified) were also removed.

Web site¹ that provides Brazilian public tenders content, with questions and answers for several subjects, including SE. To download the content, we built a second custom software that parse the pages and save the information in a JSON file that have the description of the question, all possible answers, the correct answer and the metadata of the set. The first step to download the questions and answers is to filter on the Web site only the questions related to information technology area and software engineering subject. After filtering the information we use the resulted URL to start parsing the HTML pages and download the information. We then retrieve all questions that the Web site filters and we generate a complete JSON file with all of them. Once the download finishes another step classifies the questions regarding their degree of agreement to the needs of our corpus. This step was based on the inclusion and exclusion criteria listed in Table III.

TABLE III
INCLUSION AND EXCLUSION CRITERIA USED TO CLASSIFY THE QUESTION ACCORDING THEIR COMPATIBILITY TO A CORPUS SET

Type of criteria	Description
Inclusion criteria	Questions that are direct classified as a list, definition or fact question.
Inclusion criteria	True or false questions.
Exclusion criteria	Questions that have some kind of image analysis in content.
Exclusion criteria	Questions with text fragments that required analysis and interpretation.

After this process, we have 4,186 questions and answers pairs that we aggregate to the corpus, improving its quality and reliability. By the end of the corpus creating phase, we include 89,198 phrases with a vocabulary composed by 30,482. Note that, it is difficult to find a source of knowledge, already treated and normalized, extracted from reliable sources in Portuguese.

¹<http://www.qconcurso.com>

In this article, besides our comparison between two neural networks approaches, we created a Portuguese corpus regarding software engineering. Due to current copyright regulations, we can not share this corpus as it was extract from books that are available at the market. Although, we made available a small part of it for analysis purpose besides the source code of the approaches executed in this study².

In order to test our QA system accuracy, we built a set of 90 testing questions. The aim of this task is to normalize all used test data, keeping the differences between the implemented approaches only on its models. The questions were classified according its types [3], [4], [22], [34], [35]. In Table IV we describe the amount of questions by their types and provide some examples.

TABLE IV
AMOUNT OF QUESTIONS BY ITS TYPE USED AS TESTING SET

Question		Examples
Type	Amount	
Fact	40	" <i>Quem originalmente propôs o modelo Espiral?</i> " " <i>Como ficou conhecido o período da década de 1960 at meados da década de 1980?</i> " " <i>Qual um exemplo da crise de software dos anos 1960?</i> "
Definition	25	" <i>Qual o papel do Engenheiro de Software?</i> " " <i>O que modelo de processo?</i> " " <i>O que projeto?</i> "
List	25	" <i>Quais são os princípios da engenharia de software?</i> " " <i>Quais são as vantagens em definir o desenvolvimento de software como um processo?</i> " " <i>Como são classificados os mitos do software?</i> "

C. Neural Networks

We implemented two QA approaches based on natural language processing paradigms and recurrent neural networks: DMN+ and SeqToSeq. Each approach need a specific formatted file for its training.

1) *DMN+*: This is one of the most widely-used algorithms for question answering tasks when addressing a natural language processing [15]–[17]. For this approach we created a training file based on our extracted corpus and it was loaded to the Input and Question module. On the input module the data was loaded by sentences to be encoded into distributed vector representations. The question module encodes the questions extracted from the training set into a distributed vector representation. Figure 2 shows some examples of how data must be prepared for the training phase. We carried out our experiment by using the algorithm implemented in python based on [16] findings.

2) *SeqToSeq*: This implementation consists of two recurrent neural networks, one to work as the encoder and another one as the decoder [19], [20], [36], [37]. On the encoder RNN we loaded the sequences throw a training file formatted with sentences by line, with that, the encoder would ideally capture the semantic summary of the input sequence. Based on this context, the decoder generates the output sequence.

...
 * Por algum motivo, os livros de engenharia de software quase sempre iniciam com o tema "crise do software".
 * Essa expressão vem dos anos 1970.
 * Mas o que é isso, afinal?
 * O software está em crise?
 * Parece que não, visto que hoje o software está presente em quase todas as atividades humanas.
 * Mas as pessoas que desenvolvem software estão em crise há décadas e, em alguns casos, parecem impotentes para sair dela.
 * Em grande parte, parece haver desorientação em relação a como planejar e conduzir o processo de desenvolvimento de software.
 * Muitos desenvolvedores concordam que não utilizam um processo adequado e que deveriam investir em algum, mas ao mesmo tempo dizem que não têm tempo ou recursos financeiros para fazê-lo.
 * Essa história se repete há décadas.
 * A expressão "crise do software" foi usada pela primeira vez com impacto por Dijkstra (1971)¹.
 * Ele avaliava que, considerando o rápido progresso do hardware e das demandas por sistemas cada vez mais complexos, os desenvolvedores simplesmente estavam se perdendo, porque a engenharia de software, na época, era uma disciplina incipiente.
 * Os problemas relatados por Dijkstra eram os seguintes:
 a) Projetos que estouram o cronograma.
 b) Projetos que estouram o orçamento.
 c) Produto final de baixa qualidade ou que não atenda aos requisitos.
 d) Produtos não gerenciáveis e difíceis de manter e evoluir. A:
 O que é Polimorfismo? A: a habilidade pela qual uma única operação ou nome de atributo pode ser definido em mais de uma classe e assumir implementações diferentes em cada uma dessas classes.
 ...

Fig. 2. Examples of how the training data must be prepared for DMN+ training

IV. RESULTS

In this section, we report the results of the experiments we have carried out to evaluate our proposed QA system. The results were obtained after the submission of the testing set to each implemented approach. The answers retrieved by them were manually analyzed and the data were evaluated and detailed above. The three key results of this experimental are:

- The approach based on improved dynamic memory networks reach better results than sequence to sequence approach.
- Question answering systems based on natural language processing can reach interesting results.
- Question answering systems can be used as a tool to support a teaching-learning process.

Figure 3 presents the results obtained by each executed approach regarding the answers classifications we made. What stands out in this figure is how the approach based on improved dynamic memory networks performed better than sequence to sequence based approach. The DMN+ implementation answered 62.2% question correctly while SeqToSeq approach answered correct 45.6% questions. DMN+ had a better performance on the other classifications either, it answered less incorrect questions than the sequence to sequence approach and had retrieved only 7.8% wrong answers, other than double wrong answers responded by SeqToSeq.

The differences between DMN+ and SeqToSeq are highlighted in Figure 4. We observe that the DMN+ approach answered correctly 31 fact questions against 21 answered by SeqToSeq. For the definition questions, DMN+ answered 14 questions correctly while SeqToSeq answered 11. When we analyze the correct answers for list questions provided by

²https://github.com/ma-calijorne/Software_Engineering_Corpus

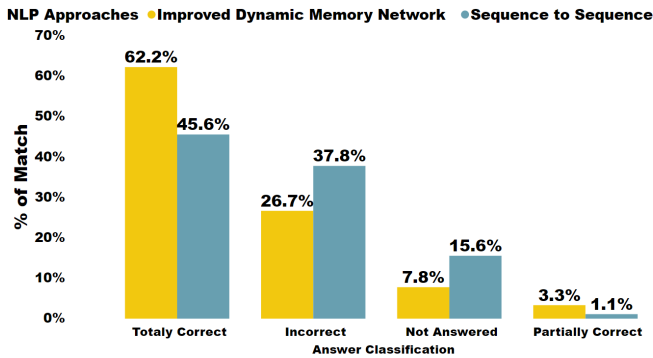


Fig. 3. Results achieved by each approach when executed with our corpus.

each approach, we can see that DMN+ answered in a correct manner 11 questions and SeqToSeq only 9 questions.

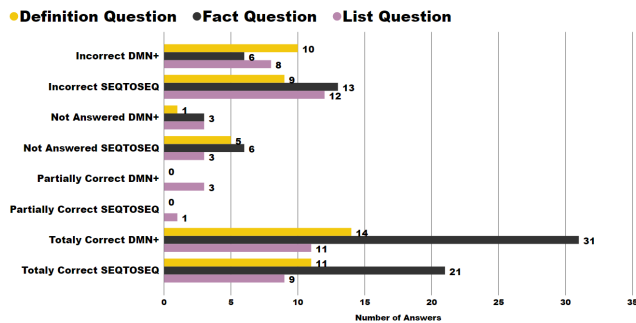


Fig. 4. Differences between DMN+ and SeqToSeq when analyzing their performances.

Regarding wrong answers provided by the approaches, we can see that SeqToSeq had a better performance on retrieving less incorrect answers when we are dealing with definition questions, on the other hand, analyzing the amount of wrong answers retrieved for fact and list questions, DMN+ had better results.

Analyzing the amount of questions that didn't retrieve any answer, we also observed that DMN+ performed better, this approach didn't answer 1 definition questions, 3 fact questions and 3 list questions, on the other hand, SeqToSeq didn't answer 5 definition questions, 6 fact questions and 3 list questions. Besides the amount of answered questions by their types, we analyzed the elapsed time on each model training and on their answer retrieval.

Regarding the time spent to train the models, Table V provides the results obtained on each model training. We can see that DMN+ spent almost 8 hours more to finish the training.

When we analyze the average elapse time classified by the question and answer types, it is possible to identify the huge difference among them. In special, the average time took by DMN+ to retrieve correct fact questions in relation to SeqToSeq.

TABLE V

TIME SPENT FOR EACH APPROACH ON MODEL TRAINING. BOTH MODELS WERE TRAINED IN THE SAME CONDITIONS OF HARDWARE AND BASIC SOFTWARE SETTINGS.

NLP Approach	Time Enlapsed (HH::MM:SS)
Improved Dynamic Memory Networks	38:26:53
Sequence To Sequence	30:37:24

V. DISCUSSION

One of the main goals of this experiment was to analyze the performance of two question answering algorithms, based on natural language processing paradigm when applied to a restrict domain. In our results we could see that improved dynamic memory networks performed better than Sequence to Sequence model, particularly as concerns when the models are answering questions of fact type. Additionally, we could see an important adherence of question answering systems to a educational environment as a tool to support a teaching-learning process.

Other authors analyzed these two models in questions answering systems as well. There are similarities between the results expressed by this work and those described by [15], [16], [18]–[20], [36]–[38], however, there is an important difference between their approaches and ours, the data used as corpus. As we detailed in section III-B, we used as a corpus, written information about software engineering while the other studies based their experiments in facebook bAbi data³ and the DATaset for QUEStion Answering on Real-world (DAQUAR).

Analyzing the results obtained by the other studies we could see that they had a better performance on answering fact questions [15], [16], [38]. We concluded that this difference of performance is related to how their corpus were prepared for the task, the data sets used by them are specially oriented for a fact question answering algorithm. Although we have created our training files based on our corpus, the nature of our data extracted from the books and training questions brought to us difficulties on mounting the tasks for each model. This issue related to the training file development lead us to embrace the DMN+ algorithm, as the substitution of the single GRU for the sentence layer and Input fusion layer allowed us to create our tasks with questions that could be answered in a segment that appears before or after it [16].

The elapse time on model training and answer retrieval was not addressed by the other works, so our study only reported our findings regarding this measure. As the creation of training files became a key task on this kind of application, more research on this topic need to be undertaken, aiming to facilitate make available QA systems for different subjects

VI. CONCLUSION AND FUTURE WORKS

In this article, we proposed a neural question answering system to support the teaching-learning process. This is the first investigation that has used question answering algorithms

³<https://research.fb.com/downloads/babi/>

● DMN+ ● SEQTOSEQ

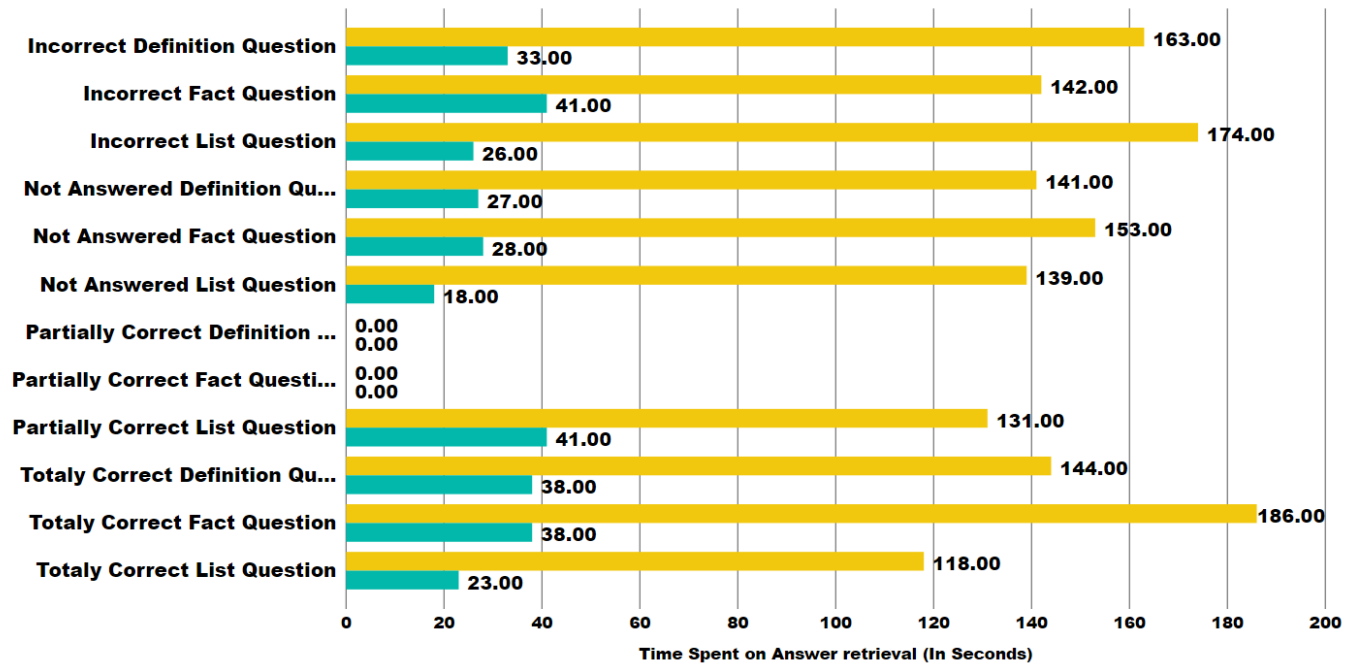


Fig. 5. Average elapse time reached by each approach classified by the answers type and the question type.

to address the Portuguese language in a system to support an educational environment. Experiments showed that in our context, improved dynamic memory networks performed better than sequence to sequence algorithm. Despite the better training and answer retrieving elapse time of SeqToSeq, their results are less interesting than the ones obtained by DMN+, particularly when we consider fact questions.

Experimental results suggest that question answering systems can be used as a tool on a teaching-learning support process. We observed a 63% of accuracy for the DMN+ algorithm when we consider all types of question. If we consider only the questions classified as fact, the performance was even better, reaching over 77% of accuracy. These research highlight the potential usefulness of question answering system in an educational environment, providing support on accurate knowledge extraction.

There is, therefore, a demand for a corpus that can be reliable and complete enough to be used as a source to the QA systems. This task is not easy and requires a major effort on retrieve and format reliable information regarding to the domain selected for the system. Further studies needs to attempt on the performance of algorithms that aims to answer complex question, the ones that need to be interpreted. There are several approaches which claims to solve this [3], [39], [40] and they should be analyzed to investigate the improvement on a tool that can be used in an educational environment. Besides that, more research on the corpus creation should be addressed to make easier the availability of QA systems for other subjects.

ACKNOWLEDGEMENTS

This work is partially funded by CNPq (Brazilian National Council for Scientific and Technological Development), MC-TIC (Brazilian Ministry of Science, Technology, Innovation and Communications), CAPES (Coordination for the Improvement of Higher Education Personnel), FAPEMIG (Foundation for Research and Scientific and Technological Development of Minas Gerais).

REFERENCES

- [1] D. Mollá and J. L. Vicedo, "Question answering in restricted domains: An overview," *Computational Linguistics*, vol. 33, no. 1, pp. 41–61, 2007.
- [2] A. Peñas, B. Magnini, P. Forner, R. Sutcliffe, Á. Rodrigo, and D. Giampiccolo, "Question answering at the cross-language evaluation forum 2003–2010," *Language resources and evaluation*, vol. 46, no. 2, pp. 177–217, 2012.
- [3] A. Ansari, M. Maknoja, and A. Shaikh, "Intelligent question answering system based on artificial neural network," in *Engineering and Technology (ICETECH), 2016 IEEE International Conference on*. IEEE, 2016, pp. 758–763.
- [4] S. P. Lende and M. Raghuvanshi, "Question answering system on education acts using nlp techniques," in *Futuristic Trends in Research and Innovation for Social Welfare (Startup Conclave), World Conference on*. IEEE, 2016, pp. 1–6.
- [5] X. Yao, "Feature-driven question answering with natural language alignment," Ph.D. dissertation, Johns Hopkins University, 2014.
- [6] Y.-g. Cao, J. J. Cimino, J. Ely, and H. Yu, "Automatically extracting information needs from complex clinical questions," *Journal of biomedical informatics*, vol. 43, no. 6, pp. 962–971, 2010.
- [7] N. Malik, A. Sharan, and P. Biswas, "Domain knowledge enriched framework for restricted domain question answering system," in *Computational Intelligence and Computing Research (ICCIC), 2013 IEEE International Conference on*. IEEE, 2013, pp. 1–7.

- [8] V. Bhoir and M. Potey, "Question answering system: A heuristic approach," in *Applications of Digital Information and Web Technologies (ICAD/IWT), 2014 Fifth International Conference on the*. IEEE, 2014, pp. 165–170.
- [9] M. Neves and U. Leser, "Question answering for biology," *Methods*, vol. 74, pp. 36–46, 2015.
- [10] S. K. Ray, S. Singh, and B. P. Joshi, "A semantic approach for question classification using wordnet and wikipedia," *Pattern Recognition Letters*, vol. 31, no. 13, pp. 1935–1943, 2010.
- [11] P. Gupta and V. Gupta, "A survey of text question answering techniques," *International Journal of Computer Applications*, vol. 53, no. 4, 2012.
- [12] S. K. Hamed and M. J. Ab Aziz, "A question answering system on holy quran translation based on question expansion technique and neural network classification," *Journal of Computer Science*, vol. 12, no. 3, pp. 169–177, 2016.
- [13] M. H. Heie, E. W. Whittaker, and S. Furui, "Question answering using statistical language modelling," *Computer Speech & Language*, vol. 26, no. 3, pp. 193–209, 2012.
- [14] D. Lukovnikov, A. Fischer, J. Lehmann, and S. Auer, "Neural network-based question answering over knowledge graphs on word and character level," in *Proceedings of the 26th international conference on World Wide Web*. International World Wide Web Conferences Steering Committee, 2017, pp. 1211–1220.
- [15] A. Kumar, O. Irsoy, P. Ondruska, M. Iyyer, J. Bradbury, I. Gulrajani, V. Zhong, R. Paulus, and R. Socher, "Ask me anything: Dynamic memory networks for natural language processing," in *International Conference on Machine Learning*, 2016, pp. 1378–1387.
- [16] C. Xiong, S. Merity, and R. Socher, "Dynamic memory networks for visual and textual question answering," in *International Conference on Machine Learning*, 2016, pp. 2397–2406.
- [17] E. Stroh, S. Student, and P. Mathur, "Question answering using deep learning," 2016, available at <https://cs224d.stanford.edu/reports/StrohMathur.pdf>.
- [18] A. M. Anything, "Dynamic memory networks for natural language processing," *Kumar et al. arXiv Pre-Print*, 2015.
- [19] K. Cho, B. Van Merriënboer, C. Gulcehre, D. Bahdanau, F. Bougares, H. Schwenk, and Y. Bengio, "Learning phrase representations using rnn encoder-decoder for statistical machine translation," *arXiv preprint arXiv:1406.1078*, 2014.
- [20] I. Sutskever, O. Vinyals, and Q. V. Le, "Sequence to sequence learning with neural networks," in *Advances in neural information processing systems*, 2014, pp. 3104–3112.
- [21] K. Vila, J.-N. Maçon, and A. Fernández, "Model-driven adaptation of question answering systems for ambient intelligence by integrating restricted-domain knowledge," *Procedia Computer Science*, vol. 4, pp. 1650–1659, 2011.
- [22] G. Nanda, M. Dua, and K. Singla, "A hindi question answering system using machine learning approach," in *Computational Techniques in Information and Communication Technologies (ICCTICT), 2016 International Conference on*. IEEE, 2016, pp. 311–314.
- [23] J. Moré, S. Climent, and M. Coll-Florit, "An answering system for questions asked by students in an e-learning context," *International Journal of Educational Technology in Higher Education*, vol. 9, no. 2, pp. 229–239, 2012.
- [24] R. Wazlawick, *Engenharia de software: conceitos e práticas*. Elsevier Brasil, 2013, vol. 1.
- [25] W. PAULA FILHO and D. PÁDUA, *Engenharia de software; fundamentos, métodos e padrões*, 2003. LTC, 2003.
- [26] K. Hirama, *Engenharia de software: qualidade e produtividade com tecnologia*. Elsevier Brasil, 2011.
- [27] I. Sommerville, "Engenharia de software-8ª edição (2007)," *Ed Person Education*, 2011.
- [28] S. R. Schach, *Engenharia de Software-: Os Paradigmas Clássico e Orientado a Objetos*. AMGH Editora, 2009.
- [29] R. S. Pressman, *Engenharia de software*. Makron books Sao Paulo, 1995, vol. 6.
- [30] M. Poppendieck and T. Poppendieck, *Implementando o desenvolvimento Lean de Software: do conceito ao dinheiro*. Bookman Editora, 2009.
- [31] R. Prikładnicki, R. Willi, and F. Milani, *Métodos ágeis para desenvolvimento de software*. Bookman Editora, 2014.
- [32] J. Sbrocco and P. Macedo, "Metodologias ágeis-engenharia de software sob medida," *São Paulo: Editora Érica Ltda*, 2012.
- [33] M. Cohn, *Desenvolvimento de software com Scrum: aplicando métodos ágeis com sucesso*. Bookman, 2000.
- [34] D. Hristovski, D. Dinevski, A. Kastrin, and T. C. Rindflesch, "Biomedical question answering using semantic relations," *BMC bioinformatics*, vol. 16, no. 1, p. 6, 2015.
- [35] S. Archana, N. Vahab, R. Thankappan, and C. Raseek, "A rule based question answering system in malayalam corpus using vibhakthi and pos tag analysis," *Procedia Technology*, vol. 24, pp. 1534–1541, 2016.
- [36] D. Bahdanau, K. Cho, and Y. Bengio, "Neural machine translation by jointly learning to align and translate," *arXiv preprint arXiv:1409.0473*, 2014.
- [37] O. Vinyals and Q. Le, "A neural conversational model," *arXiv preprint arXiv:1506.05869*, 2015.
- [38] A. Raghuvanshi and P. Chase, "Dynamic memory networks for question answering," 2016, available at <https://cs224d.stanford.edu/reports/RaghuvanshiChase.pdf>.
- [39] Y. Chali, S. A. Hasan, and M. Mojahid, "A reinforcement learning formulation to the complex question answering problem," *Information Processing & Management*, vol. 51, no. 3, pp. 252–272, 2015.
- [40] S. Pudaruth, K. Boodhoo, and L. Goolbudun, "An intelligent question answering system for ict," in *Electrical, Electronics, and Optimization Techniques (ICEEOT), International Conference on*. IEEE, 2016, pp. 2895–2899.