Arabic Question-Answering System Using Search Engine Techniques

Manal Alamir[1], Sadeem Alharth[1], Shahad Alqurashi[1],
and Tahani Alqurashi[2](✉)

[1] College of Computer and Information Systems, Umm Al Qura University,
Makkah, Saudi Arabia
{s44180009,s44181412,s44180184}@st.uqu.edu.sa

[2] Common First Year Deanship, Umm Al-Qura University, Makkah, Saudi Arabia
tmqurashi@uqu.edu.sa

Abstract. The Arabic language is one of the most widely spoken languages in the world. Many natural language processing experts have tried to understand its linguistic complexity. This makes text processing and its applications difficult, particularly in question-answering systems. Some researchers have unsuccessfully tried to tackle the problem of creating an effective question-answering system. In this paper, we present a question-answering system for a Saudi Arabia labor law dataset. Our system works in three main stages named Data Preparation, Data Preprocessing and Answer Extraction. The main aim of the first two stages is to prepare and preprocess the dataset in order to be in a suitable format for building question-answering system. In the Answer Extraction stage, two text similarity measurements are applied, which is TF-IDF and Cosine. Then, the candidate answers are evaluated and ranked based on their similarity scores and the most relevant answer to the user's query is displayed as a final answer. We evaluated our proposed system by test it using 100 of manually generated user queries and on average we achieved a good results.

Keywords: Arabic natural language processing · Question answering · Search query · Relevant information

1 Introduction

The Internet and related applications have become the main source of information for experts, researchers, students, and the general user [4]. However, these applications contain vast amount of information, and sometimes a query requires a specific answer. The user often must spend excessive time searching the list of retrieved documents or information to find the answer they are looking for. Thus, an application-based search engine would be useful. In addition, most of the documents available for retrieval on the internet are written in English. Many

W. Fu et al. (Eds.): ICMTEL 2021, LNICST 387, pp. 333–343, 2021.
https://doi.org/10.1007/978-3-030-82562-1_31

users need access to papers in other languages, specifically Arabic since a large part of the world's population speak the language [12].

For these reasons, we built a search engine system that caters to Arabic ministries, specifically the System of Labor in the Ministry of Human Resources and Social Development. The System of Labor contains many legal documents and articles concerning many legal details. Government employees and law personnel must currently spend a great deal of time searching the list of retrieved articles or legal documents to find related answers because they are not arranged in a way that makes it easy to obtain information. As we will show in the literature review section, different techniques have been deployed in the creation of question-answering systems. However, free form-based search engines are the most effective means of achieving the objective of the system used here in which the content of a chapter (text) is searched and retrieve back its number and text as an answer to the user. This data was then organized into a free form-based format that can be easily accessed publically. Thus, in this paper, we present a way to organize the data of the System of Labor to be easily retrievable for any user. The rest of this paper illustrates our methodology (Sect. 3) and presents the results that we obtained (Sect. 4).

2 Related Work

There is a difference between a search engine system and a question-answering system. Search engine systems provide query results using databases that have already indexed documents [8]. The typical search engine is not designed to extract specific answers to queries and returns a set of references that may contain the answers; a question-answering system retrieves direct or correct answers to the questions rather than flooding the user with documents or giving general answers [11].

Research in the field of Arabic language processing is very limited, and this is one of the challenges that we faced in the implementation of our project. AlMaayah et al. [5] offered an approach for extracting the synonyms of words in the Qur'an, specifically Juzo Amma. They used the WordNet dataset, which depends on traditional Arabic dictionaries and a collection of techniques, such as frequency and inverse document frequency, cosine similarities, and three measures: precision, recall, and f-measure. Their search performance reached an accuracy of 27%.

Malhas et al. [10] provided (AyaTEC) test set contents 207 questions verse-based on the Holy Qur'an for question answering system. That presented information needs of both skeptical users and inquisitive which covered eleven subject classes of the Holy Qur'an. They made AyaTEC set available to the research society to develop this field further. They proposed many evaluation measures to backing the kind of verse-based answers and questions systems while merging in the evaluation the notion of partial matching of answers. In many instances, the search is incapable to call pertinent verses where it did not use the semantic relation between words in the query.

Recently, Zouaoui et al. [23] introduced an ontology-based semantic search engine as an index and acquired good results in terms of precision and recall measures. The search engine based on a collection of beneficial terms extracted from the Quranic Earab book with grammatical functions that avail as definitions, they concentrate on constructing a new ontology for the Quranic text to use for information retrieval. While, Zeid et al. [21] used graph ontology together with a web search API as an alternate track to get answers and perform the ontology.

Hani et al. [3] used an ontological resource to match their search with semantically similar words. They divided the processing of the query as follows: first they extracted an enhanced query using keywords and then passed those results through an ontological resource to get semantically correlated words. They tested this approach on 50 Arabic questions from the standard set of TREC and CLEF Arabic questions and received a mean reciprocal ratio for retrieving documents of 1.53, which is considered good.

Paolo, et al. [13] used a regular question-answering system that had three modules: a question analysis module, a passage retrieval module, and an answer extraction module. To be able to analyze the question, they implemented their own Arabic Named Entity Recognition system to determine the named entities in the question that will determine the type of answer. They adapted an already implemented system called JIRS for their passage retrieval module. When tested on a dataset of 200 questions, the overall performance of their model was good.

Jaffar et al. [6] provided a query expansion review in Arabic. This was basically divided into three strategies. The first used classical methods of stemming, lemmatization, and word sense disambiguation. The second strategy used a feedback score to find matching results for query enhancement. The third strategy used data extracted from a corpus or external resources such as WordNET.

Dima et al. [19] proposed a system that extracts keywords using word2vec. In their system, the text was first pre-process to word stems. The stems were used to calculate the bag of concepts used to make a linear combination of each word that shows the concepts that define it. Next, the words were catagorized to similar classes to determine synonyms and similar words. They then calculated the different N-gram weights to extract the keywords.

Dima et al. [18] presented a study on the usage of word embeddings in the Arabic language. They showed the effect of the word embeddings used from GLOVE and Skip-gram and CBOW. The study showed that the applications included sentiment analysis and classification of sentiment and that they are applied in measuring semantic-based similarities. The applications also included short-answer grading and information-retrieval applications, such as cosine similarities. The similarity-based applications included plagiarism detection and paraphrasing identification.

Wissam, et al. [20] have tested five web search engines on an information retrieval evaluation for fifty queries chosen randomly essential with the observance of the web-specific estimate requirements. The descriptions of the top ten results and relation they're for all queries were evaluated by independent jurors.

The essence return was that Google perfected roughly all the times better than the other different engines.

Majed, et al. [15] have presented explain and identify the restrictions of the difficulties of Arabic documents retrieving. They approach used three Arabic search engines: Yahoo, Google, and Idrisi. They applied stemming and spelling normalization techniques were not enough techniques requirement significantly develop retrieval, when retrieval by n-grams technique was more efficient for indexing Arabic documents.

Zhong et al. [22] sophisticated approach to retrieve questions pertaining to building regulations. The approach combine deep learning model of Natural Language Processing (NLP) with information retrieval to get specific and fast answers to user's queries about building regulations. They proposed a chatbot System for building regulations question answering.

3 Arabic Question Answering System

Figure 1 shows the architecture of our proposed Arabic Question Answering System. The system consists of three main stages, which are Data Preparation, Data Preprocessing and Answer Extraction. The following subsections explain them in more details:

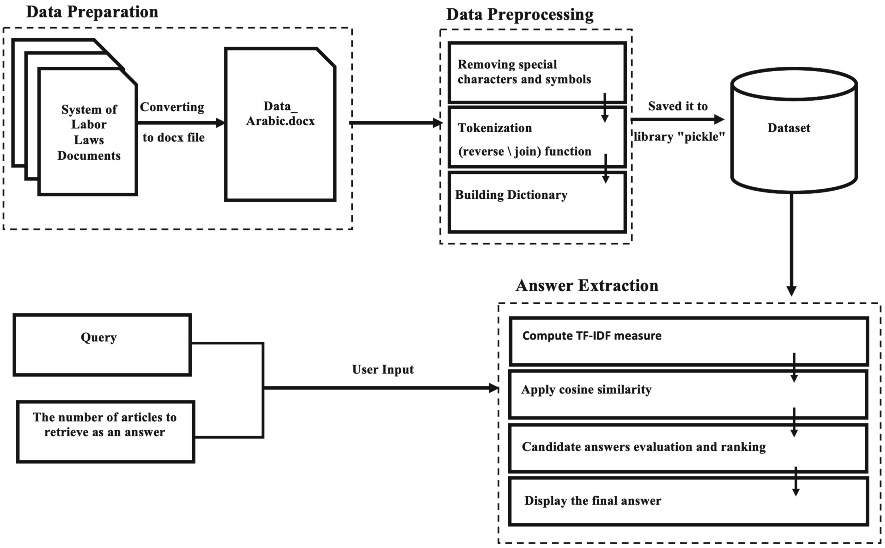


Fig. 1. The Arabic question answering system architecture.

3.1 Data Preparation

For this project, we worked with the labor law dataset available on the website of the Ministry of Human Resources and Social Development in the Kingdom of Saudi Arabia [2]. The dataset contained 245 articles divided into six sections. This type of dataset is known as primary data, which refers to data collected from first-hand experience. Primary data has not yet been published and is more reliable, authentic, and objective.

The data file was originally received in PDF format, and it was converted to a docx file format to facilitate the use of the Python language; unfortunately, when handling Arabic text, Python packages often destroy much of the information. We used docx2txt, which is a pure python-based utility, to read text from Microsoft Word documents or extract images from docx files [1]. Docx2txt also performs a fundamental function of converting data inside the docx files into rows, which enables the mining and programming of data. The next section presents the data preprocessing steps in detail.

3.2 Data Preprocessing

After using the docx2txt library, we were surprised at the amount of preprocessing that the data still needed, as it contained many characters, punctuation marks, and spaces that would have made the query processing difficult; the texts also appeared in reverse, as shown in Fig. 2. Thus, several steps were used to organize the data.

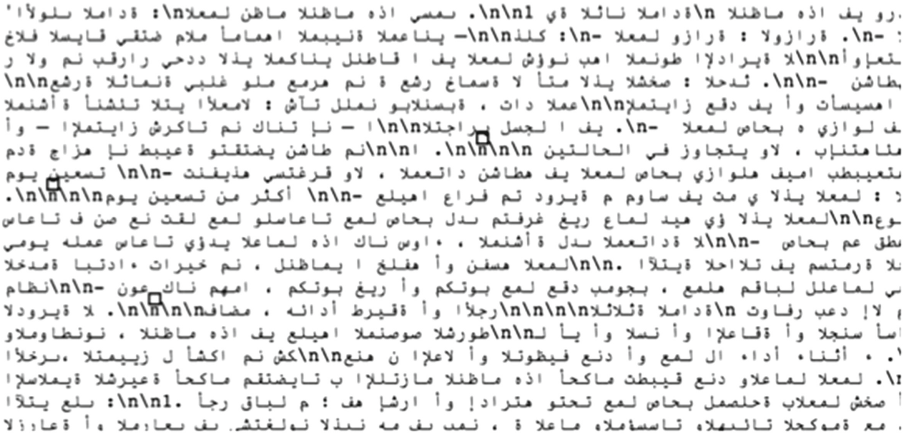
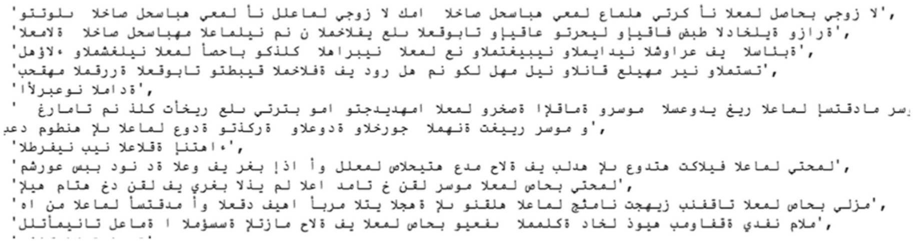


Fig. 2. Outputs before preprocessing stage

First, strip function with separator `\n` was used to remove specified spaces at the beginning and end of the text; Fig. 2 shows that the text was still in need of further processing.

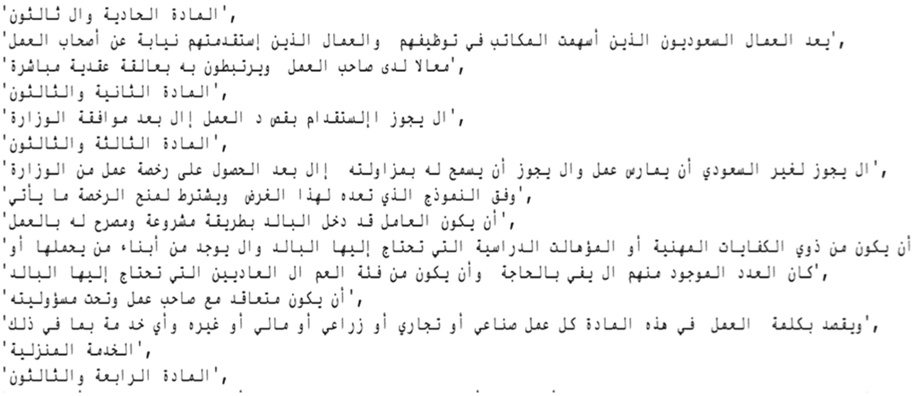
Next, regular expressions were used to remove any special characters, spaces, punctuations, and English letters, such as $, ∗, xa0, ...etc. Figure 3 shows the output after this stage.



```
'لا زوجي بحاصل لمعلا نأ كرتي هلماع لمعي هباسحل صاخلا امك لا زوجي لماعلل نأ لمعي هباسحل صاخلا ىلوتو',
'ةرازو ةيلخادلا طبض فاقيإو ليحرتو عاقيإو تابوقعلا ىلع يفلاخملا ن نم نيلماعلا مهباسحل صاخلا ةلامعلا',
'ةبئاسلا يف عراوشلا نيدايملاو نيبيغتملاو نع لمعلا نيبراهلا كلذكو باحصأ لمعلا نيلغشملاو ءلاؤهل',
'تسملاو نير مهيلع قانلاو نيل مهل لكو نم هل رود يف ةفلاخملا قيبطتو تابوقعلا ةررقملا مهقحب',
'ةداملا نوعبرلأا',
رس مادقتسإ لماعلا ريغ يدوعسلا موسرو ةماقلإا ةصخرو لمعلا امهديدجتو امو بترتي ىلع ريخأت كلذ نم تامارغ '
و موسر ريبغت ةنهملا جورخلاو ةدوعلاو ةركذتو ةدوع لماعلا ىلإ هنطوم دعب',
'ءاهتنإ ةقلاعلا نيب نيفرطلا',
'لمحتي لماعلا فيلاكت هتدوع ىلإ هدلب يف ةلاح مدع هتيحلاص لمعلل وأ اذإ بغر يف وعلا ةد نود ببس عورشم',
'لمحتي بحاص لمعلا موسر لقن خ تامد اعلا لم يذلا بغري يف لقن دخ هتام هيلإ',
'مزلي بحاص لمعلا تاقفنب زيهجت نامثج لماعلا هلقنو ىلإ ةهجلا يتلا مربأ اهيف دقعلا وأ مدقتسأ لماعلا نم اه',
'ملام نفدي ةقفاومب هيوذ لخاد ةكلمملا نفعيو بحاص لمعلا يف ةلاح مازتللإا ةسسؤملا ا ةماعل تانيمأتلل',
```

Fig. 3. Outputs after preprocessing

As Fig. 3 shows, the text was in reverse order, so to put it in the correct order, we looped on each word separately and reversed its characters. The output of this stage is shown in Fig. 4.



```
'المادة الحادية وال ثالثون',
'يعد العمال السعوديون الذين أسهمت المكاتب في توظيفهم والعمال الذين إستقدمتهم نيابة عن أصحاب العمل',
'معالا لدى صاحب العمل ويرتبطون به بعالقة عقدية مباشرة',
'المادة الثانية والثالثون',
'ال يجوز االستقدام بقص د العمل إال بعد موافقة الوزارة',
'المادة الثالثة والثالثون',
'ال يجوز لغير السعودي أن يمارس عمل وال يجوز أن يسمح له بمزاولته إال بعد الحصول على رخصة عمل من الوزارة',
'وفق النموذج الذي تعده لهذا الغرض ويشترط لمنح الرخصة ما يأتي',
'أن يكون العامل قد دخل البالد بطريقة مشروعة ومصرح له بالعمل',
أن يكون من ذوي الكفايات المهنية أو المؤهالت الدراسية التي تحتاج إليها البالد وال يوجد من أبناء من يحملها أو'
'كان العدد الموجود منهم ال يفي بالحاجة وأن يكون من فئة العم ال العاديين التي تحتاج إليها البالد',
'أن يكون متعاقد مع صاحب عمل وتحت مسؤوليته',
'ويقصد بكلمة العمل في هذه المادة كل عمل صناعي أو تجاري أو زراعي أو مالي أو غيره وأي خد مة بما في ذلك',
'الخدمة المنزلية',
'المادة الرابعة والثالثون',
```

Fig. 4. Final outputs

3.3 Answer Extraction

After preprocessing the data, the user is asked to enter the query into our system, and it should be in the form of asking about the article that is related to specific topic along with the number of articles that he/she wish to retrieve as an answer k.

Then we apply two text similarity measurements to calculate the similarity between each words in the user query and our dataset as follows:

The Term Frequency-Inverse Document Frequency (TF-IDF). Next, we computed the term frequency-inverse document frequency (TF-IDF) for the terms mentioned in our search engine. The idea was to combine the frequency of a term in a context (query) with its relative frequency in the documents overall. According to Salton and Buckley [14], the TF-IDF weighs a term's frequency (TF) and its inverse document frequency (IDF).

To calculate TF, we built the inverted index representation which consists of two basic components: the inverted list and the vocabulary. The inverted list is the reference about the place of that word in the documents set, its frequency, position,... etc. and it is represented by a vector. While, the vocabulary is the one word from the document set. In the index, each word has an inverted list include reference of occurrences of the word in special documents [4,16].

Then we built the inverted index of our system, in which each word mentioned is connected with the chapter it is mentioned in. Inverted index representation is the main building block for any search engine or information retrieval system and allows for the calculation of the text similarity metrics.

After we calculated TF, we computed the TF-IDF using the following formulas:

$$\begin{aligned} &\text{TF-IDF} = TF(i,j) * IDF(i) \\ &TF(i,j) = \frac{\text{Frequency of word i in document j}}{\text{Total words in document j}} \\ &IDF(i) = \log(\frac{\text{Total documents}}{\text{documents with word i}}) \end{aligned} \tag{1}$$

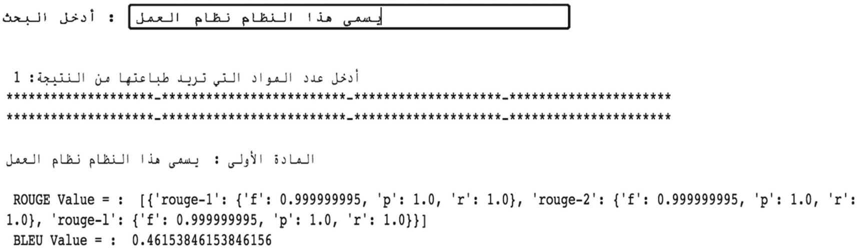
Cosine Similarity Function. We were then able to compute the similarity functions needed for information retrieval based on the dictionary using the TF-IDF scores for our system. This stage includes two types of computations. First, we summed up the similarities of each word of the query with each document based on the morphology and semantics of the word. We also calculated the length of each of the queries and the length of the chapter's text: this information is needed for the next type of computation, which involves calculating the cosine similarity between the input query from the user and the candidate texts. Since cosine similarity represent the projection of one word vector to another, it was used to measure the whole projection of the word vector (the query) and the text of the chapter in the dataset. The use of this type of similarity proved to be very efficient and useful for our results, which are presented in the next section. After calculation of the similarity metrics with all the chapters available in our data, we ranked the documents in a descending order, and we displayed the *k* most relevant documents to the whole user's query vector.

4 Experiment Results

In the final stage of this study, we tested the system performance using 100 user queries and calculated both the BLEU [7] and ROUGE [9] as an evaluation measures for each query. BLEU and ROUGE measures were used as they are the most popular methods for testing the performance of machine translation, summarization, and question-answering models. BLEU and ROUGE ranged from 0 to 1, where 0 value indicates a perfect result of the mismatching text, while 1 indicates a perfect result for matched text. ROUGE results contain three symbols: f, p, and r where it refers to f1-score, precision, and recall percentage respectively. Recall simply means how much of the user queries is the model capturing or recovering. In other words, recall (r) means is the number of overlapping words appearing in both model and user queries divided by the number of words in user queries [17]. While precision (p) means is the number of words appearing in both model and user queries divided by the number of words in the model [17]. It is always best to compute both the recall and precision in ROUGE and then report the f1-score [9]. F1-score calculation is done by using the following equation.

$$\mathrm{F} - score = \frac{2^{*}\mathrm{p}^{*}\mathrm{r}}{\mathrm{p+r}} \tag{2}$$

However, we used ROUGE measure on unigrams level (ROUGE-1) and biograms level (ROUGE-2). Also, high accuracy was indicated using the BLEU measurement, which indicates a high accuracy when more words match between the result and the entered query. Figures 5 and 6 show some of the user queries tested against the model and their corresponding scores.



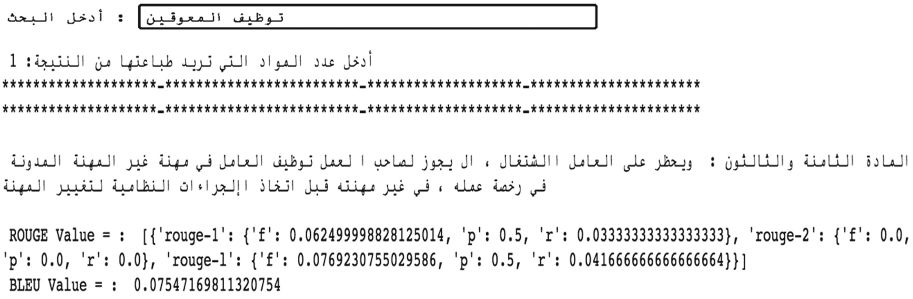
```
أدخل البحث :  يسمى هذا النظام نظام العمل

 أدخل عدد المواد التي تريد طباعتها من النتيجة: 1
*******************_************************_*******************_*********************
*******************_************************_*******************_*********************

المادة الأولى :  يسمى هذا النظام نظام العمل

 ROUGE Value = :  [{'rouge-1': {'f': 0.999999995, 'p': 1.0, 'r': 1.0}, 'rouge-2': {'f': 0.999999995, 'p': 1.0, 'r':
1.0}, 'rouge-l': {'f': 0.999999995, 'p': 1.0, 'r': 1.0}}]
 BLEU Value = :  0.46153846153846156
```

Fig. 5. The result of query that has a high similarity with the text in the dataset.

Figure 5 shows that the highest values of measures ROUGE-1, ROUGE-2 and BLEU are achieved when there is a high percentage of similarity between the user's query and the text in our dataset.

Exactly the contrary in Fig. 6, we believe that these lower values of F, P, and R do not mean mismatching between the user's query and the result, but the number of entered words by the user is much less than the number of result's words. This impacts the scores because the similarity of words is divided by the total number of words in the user's query or results as we have mentioned in P and R laws.

Figure 7 shows the average results of BLEU, ROUGE-1 and ROUGE-2 of the 100 user queries. However, on average our proposed system performance nearly



```
أدخل البحث : توظيف المعوقين

أدخل عدد المواد التي تريد طباعتها من النتيجة: 1
*******************_************************_*******************_*********************
*******************_************************_*******************_*********************

المادة الثامنة والثالثون : ويحظر على العامل االشتغال ، ال يجوز لصاحب ا لعمل توظيف العامل في مهنة غير المهنة المدونة
في رخصة عمله ، في غير مهنته قبل اتخاذ اإلجراء ات النظامية لتغيير المهنة

ROUGE Value = :  [{'rouge-1': {'f': 0.062499998828125014, 'p': 0.5, 'r': 0.03333333333333333}, 'rouge-2': {'f': 0.0,
'p': 0.0, 'r': 0.0}, 'rouge-l': {'f': 0.0769230755029586, 'p': 0.5, 'r': 0.041666666666666664}}]
BLEU Value = :  0.07547169811320754
```

Fig. 6. The result of query that has a moderate similarity with the test in the dataset.

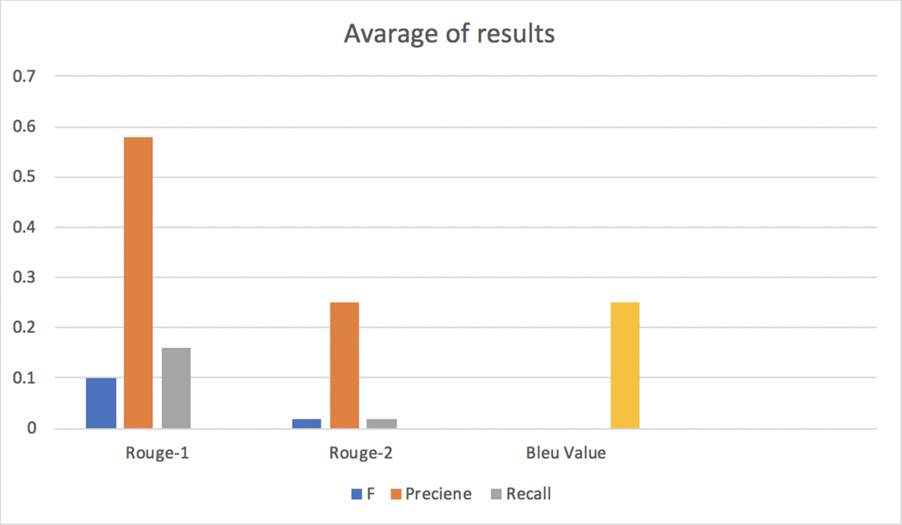


Fig. 7. The average results to evaluate 100 queries.

reached 0.6 using ROUGE-1 (P), and above 0.3 using ROUGE-2 (P) and BLEU measurements.

In summary, we can say that our system helps people who are familiar with the words used in the documents of Ministry of Human Resources and Social Development to get their desired answer in an effective way.

5 Conclusion and Future Work

We implemented an automatic approach for the search engine of the Ministry of Labor system of the Ministry of Human Resources and Social Development in the Kingdom of Saudi Arabia. We created a dataset from the original document from the website of the Ministry of Labor, and we estimated the similarity between words in a query and each chapter in the document. The more relevant documents were retrieved as an answer to the user. A limitation of the proposed System is that the form of the words in the query must have some similarity as the words in the Ministry of Labor system's dataset to return a high accurate results and this is acceptable at this stage as most user of the Ministry of Labor systems are more familiar with the vocabulary used in its documents.

Future work should include developing a question-answering system that deals with the semantics of words in order to improve the accuracy of a meaning search of a query, creating a dialogue between the user and the system in order to retrieve information more specific to the Ministry of Labor system, and expanding the task for use with more than one system.

References

1. PyPI. https://pypi.org/project/docx2txt/
2. The system of labor, the ministry of human resources and social development, in the kingdom of Saudi Arabia. https://hrsd.gov.sa/ar/policies
3. Al-Chalabi, H., Ray, S., Shaalan, K.: Semantic based query expansion for arabic question answering systems. In: 2015 First International Conference on Arabic Computational Linguistics (ACLing), pp. 127–132. IEEE (2015)
4. Al-Jedady, A.A., Al-Kabi, M.N., Alsmadi, I.M.: Fast arabic query matching for compressed arabic inverted indices. Int. J. Database Theory Appl. **5**(4), 81–94 (2012)
5. AlMaayah, M., Sawalha, M., Abushariah, M.A.: Towards an automatic extraction of synonyms for quranic arabic wordnet. Int. J. Speech Technol. **19**(2), 177–189 (2016)
6. Atwan, J., Mohd, M.: Arabic query expansion: a review. Asian J. Inf. Technol. **16**(10), 754–770 (2017)
7. Callison-Burch, C., Osborne, M., Koehn, P.: Re-evaluation the role of bleu in machine translation research. In: 11th Conference of the European Chapter of the Association for Computational Linguistics (2006)
8. Laurent, D., Séguéla, P., Nègre, S.: Qa better than IR? In: Proceedings of the Workshop on Multilingual Question Answering-MLQA 2006 (2006)
9. Lin, C.Y.: Rouge: a package for automatic evaluation of summaries. In: Text Summarization Branches Out, pp. 74–81 (2004)
10. Malhas, R., Elsayed, T.: Ayatec: building a reusable verse-based test collection for arabic question answering on the holy qur'an. ACM Trans. Asian Low Resour. Lang. Inf. Process. **19**, 781–7821 (2020)
11. Mervin, R.: An overview of question answering system. Int. J. Res. Adv. Technol. (IJRATE) **1** (2013)
12. Moukdad, H.: Lost in cyberspace: How do search engines handle arabic queries? In: Proceedings of the Annual Conference of CAIS/Actes du congrès annuel de l'ACSI (2004)
13. Rosso, P., Benajiba, Y., Lyhyaoui, A.: Towards an arabic question answering system (2006)
14. Salton, G., Buckley, C.: Term-weighting approaches in automatic text retrieval. Inf. Process. Manag. **24**(5), 513–523 (1988)

15. Sanan, M., Rammal, M., Zreik, K.: Internet arabic search engines studies. In: 2008 3rd International Conference on Information and Communication Technologies: From Theory to Applications, pp. 1–8. IEEE (2008)
16. Scholer, F., Williams, H.E., Yiannis, J., Zobel, J.: Compression of inverted indexes for fast query evaluation. In: Proceedings of the 25th Annual International ACM SIGIR Conference on Research and Development in Information Retrieval, pp. 222–229 (2002)
17. Steinberger, J., Ježek, K.: Evaluation measures for text summarization. Comput. Inf. **28**(2), 251–275 (2012)
18. Suleiman, D., Awajan, A.: Comparative study of word embeddings models and their usage in arabic language applications. In: 2018 International Arab Conference on Information Technology (ACIT), pp. 1–7. IEEE (2018)
19. Suleiman, D., Awajan, A.A., Al Etaiwi, W.: Arabic text keywords extraction using word2vec. In: 2019 2nd International Conference on new Trends in Computing Sciences (ICTCS), pp. 1–7. IEEE (2019)
20. Tawileh, W., et al.: Evaluation of five web search engines in arabic language. In: LWA, pp. 221–228 (2010)
21. Zeid, M.S., Belal, N.A., El-Sonbaty, Y.: Arabic question answering system using graph ontology. In: Silhavy, R., Silhavy, P., Prokopova, Z. (eds.) CoMeSySo 2020. AISC, vol. 1294, pp. 212–224. Springer, Cham (2020). https://doi.org/10.1007/978-3-030-63322-6_17
22. Zhong, B., He, W., Huang, Z., Love, P.E., Tang, J., Luo, H.: A building regulation question answering system: a deep learning methodology. Adv. Eng. Inf. **46**, 101195 (2020)
23. Zouaoui, S., Rezeg, K.: A novel quranic search engine using an ontology-based semantic indexing. Arab. J. Sci. Eng. **46**, 1–22 (2021)