IJCDS 1571013400

International Journal of Computing and Digital Systems
ISSN (2210-142X)
Int. J. Com. Dig. Sys. #, No.# (Mon-20..)

A New Semantic Search Approach for the Holy Quran Based On Discourse Analysis and Advanced Word Representation Models

Samira LAGRINI[1], Amina DEBBAH[1], and Mohammed REDJIMI[2]

[1] *Computer Science Department, Badji Mokhtar University, Annaba, Algeria*
[2] *Computer Science Department, University 20 Aout 1955, Skikda, Algeria*

Abstract: Semantic search is an information retrieval technique that seeks to understand the contextual meaning of words to find more accurate results. It remains an open challenge, especially for the Holy Quran, as this sacred book encodes crucial religious meanings with a high level of semantics and eloquence beyond human capacities. This paper presents a new semantic search approach for the Holy Quran. The presented approach leverages the power of contextualized word representation models and discourse analysis to retrieve semantically relevant verses to the user's query, which do not necessarily appear verbatim in Quranic text. It consists of three crucial modules. The first module concerns the discourse segmentation of Quranic text into discourse units. The second module aims to identify the most effective word representation model for mapping the Quranic discourse units into semantic vectors. To this end, the performance of five cutting-edge word representation models in assessing semantic relatedness in the Holy Quran at verse level is investigated. The third module concerns the semantic search model. Evaluation results of the proposed approach are very promising. The average precision and recall are 90.79% and 79.57%, respectively, which demonstrates the strength of the proposed approach and the ability of contextualized word representation models to capture Quran semantic information.

Keywords: Information retrieval, Natural Language Processing, contextualized word representation models, discourse analysis, semantic relatedness, Artificial Intelligence, Holy Quran.

1. INTRODUCTION

Semantic relatedness is a Natural Language Processing (NLP) task that consists of assessing the level of relatedness between two text units in a given language [1]. Usually, semantic relatedness is employed synonymously with semantic test similarity. However, semantic relatedness considers a broader view by analyzing the common semantic properties of two words [1]. For instance, the words “milk” and “cup” are considered semantically related but not semantically similar, whereas the words “diabetes” and “illness” are semantically similar. This is why semantic relatedness is considered a broader area encompassing semantic similarity. In the present research, using the term ‘semantic relatedness’ is more accurate.

Assessing the semantic relatedness between text units plays an essential role in various NLP tasks, including information retrieval [2] and, more precisely, semantic search. Semantic search is an information retrieval technique that seeks to find the inherent meanings of words and their semantic relationships to accurately retrieve relevant documents to the user’s query [3]. It is dissimilar to lexical search, where the retrieval system searches for the lexical matching of query words, ignoring their contextual meaning and semantic relationships. A semantic search system should be able to understand the user’s query and accurately find its semantically related documents that describe the query subject, even if a lexical match of query words is not found. There is a persistent need for such system for the Holy Quran to search semantically related verses that discuss a specific topic. The Holy Quran is the most sacred religious book for more than 2 billion Muslims and their source of guidance. It’s an exceptional Classical Arabic (CA) text with an incomparable description and meaning style [4]. This sacred book addresses all relevant subjects for people and details refined spiritual meanings in a specific manner that can be revealed by a broad

E-mail address: samira.lagrini@univ-Annaba.dz, amina.debah@ univ-Annaba.dz, m.redjimi@univ-skikda.dz

analysis [5]. This is why Muslims and even non-Muslims try to understand its content, and deeply analyze its related verses to get an accurate and broad explanation of the intended meanings. Studying all semantically related verses together gives us an exhaustive idea about the target subject, and helps us gain in-depth religious knowledge and understand several judgments in this sacred book.

However, developing a semantic search system for the Holy Quran is primarily a challenging task due to the following exceptional features of this sacred book:

Firstly, when analyzing the Holy Quran, we notice that semantically related verses generally discuss the same subject without referring lexically to this subject. Taking as an example the following two verses:

(1)« خُشَّعًا أَبْصَارُهُمْ يَخْرُجُونَ مِنَ الْأَجْدَاثِ كَأَنَّهُمْ جَرَادٌ مُنْتَشِرٌ »
(القمر، 7)

English Translation: "Their eyes humiliated, they will emerge from the graves, as if they were swarming locusts." (Al-Qamar 7)

(2) «مُهْطِعِينَ مُقْنِعِي رُءُوسِهِمْ لَا يَرْتَدُّ إِلَيْهِمْ طَرْفُهُمْ ۖ وَأَفْئِدَتُهُمْ هَوَاءٌ»
(ابراهيم، 43)

English Translation: "Scrambling with their heads upturned, there will be a fixed gaze in their eyes and their hearts will be vacant." (Abraham,43).

Both verses are semantically related. They address the same subject, 'the situation of people on judgment day', but without any explicit reference to this subject. Any lexical search system based on lexical matching of keywords is not able to detect that these verses are relevant and strongly related to the subject 'judgment day'.

Secondly, most verses in the Holy Quran, especially long verses, tackle several topics at the same time. For instance, in example (3), we notice that the verse discusses three topics through its discourse units (between '[]'): the first one talks about the creatures, while the second concerns the comprehensiveness of the sacred book, and the last is about the banishment. This feature suggests that the same verse could be relevant to several topics and that it may also be semantically related to other verses that deal with at least one of its discussed topics.

(3) [وَما مِن دابَّةٍ فِي الأَرضِ وَلا طائِرٍ يَطيرُ بِجَناحَيهِ إِلّا أُمَمٌ أَمثالُكُم ۚ]$_1$
[ما فَرَّطنا فِي الكِتابِ مِن شَيءٍ ۚ]$_2$[ثُمَّ إِلىٰ رَبِّهِم يُحشَرون]$_3$
(الانعام 38)

English Translation: [There is no animal on land, nor a bird that flies with its wings, but they are communities like yourselves.]$_1$ [We have not omitted anything from the Book.]$_2$ [Then they will be mustered toward their Lord.]$_3$ (Cattle, 38)

However, when Quranic verses address one single topic, we notice that the main topic is generally stated in a single discourse unit, while the remaining parts of the verse (discourse units) provide further information about the main topic, either an explanation (cf. example 4) or a consequence. To name a few.

(4) [هٰذَا كِتَابُنَا يَنطِقُ عَلَيْكُم بِالْحَقِّ ۚ]$_1$[إِنَّا كُنَّا نَسْتَنسِخُ مَا كُنتُمْ تَعْمَلُونَ]$_2$
(الجاثية, 29)

English Translation: [This is Our book, which speaks truly against you]$_1$[Indeed We used to record what you used to do] $_2$ (Crowling, 29)

Considering such features is primordial when developing a semantic search system for the Holy Quran. However, existing search tools completely neglect the Quran discourse and its particularities. Most of these tools are based on classical information retrieval techniques, either on lexical matching of query words or on the use of ontology and word synonyms, which are time-consuming techniques and involve rich linguistic resources.

In this paper, a new semantic search approach for the Holy Quran is proposed. The presented approach aims to overcome the limitations of the existing approaches by relying mainly on Quran discourse segmentation and advanced word representation models. Our goal is to improve the accuracy of semantic search in the Holy Quran by leveraging the power of advanced word representation techniques to capture the semantic relatedness in the Holy Quran, as well as properly target the covered topics in verses via our original discourse segmentation method. The three main contributions of this paper are discussed below:

1. A new semantic search approach for the Holy Quran is presented. The presented approach overcomes the limitations of existing approaches and accurately detects semantically related verses to the user's input query.

2. The paper investigates the effectiveness of advanced word representation models, trained on a classical Arabic corpus, for semantic relatedness in the Holy Quran at the verse level, and discusses the obtained findings. To the best of our knowledge, this is the first research work that tackles this problem using several advanced word representation models.

3. This research presents an original method for Quran discourse segmentation that ensures the coherence of the generated discourse units and provides a comprehensive insight for researchers interested in developing new ideas when tackling the issue of semantic search in the Holy Quran.

The remainder of the paper is organized as follows: Section 2 briefly reviews recent research carried out on the Holy Quran; emphasis is put on semantic relatedness

and semantic search. Section 3 presents the proposed approach and its main steps. Section 4 reports the experiments and evaluation results. Finally, Section 5 concludes the paper and mentions some future directions.

2. RELATED WORK

Recently, the Holy Quran has gained significant attention and has been considered a hot research subject in NLP. Several types of research have been conducted on the Holy Quran. In this section, we provide a brief review of previous research on semantic relatedness and semantic search over the last decade.

One interesting research conducted on the Holy Quran is QurSim [6], an interesting resource of semantically related verses. QurSim contains 7679 pairs of related verses, annotated with three levels of relatedness according to Ibn Kathir's interpretation. To evaluate the similarity among pairs of verses in QurSim, Term Frequency- Inverse Document Frequency (TF-IDF) and cosine distance [7] were employed. The authors improved their corpus by incorporating Quran anaphoric information [8].

Authors in [9] used TF-IDF technique to compute similarity among verses in the Holy Quran. The authors only considered shared words to find the most similar verses to the user's query verse. This work was extended by performing binary classification of Quran chapters into 'Makki' and 'Madani' classes using N-gram and LibSVM classifiers.

In [4], the authors proposed a multi-corpus vector space model to estimate the semantic relatedness among verses in the Holy Quran. Each verse has undergone two levels of representation. The first level used the 'Qurana' corpus to expand verses' representation by the shared concepts. The second level used a list of synonyms, collected using Arabic online dictionaries, to enrich the verse's vector. To compute the similarity among each pair of verses, the cosine measure was used.

In the same context, authors in [5] investigated the use of Doc2vec model and cosine similarity measure to detect semantically related verses in the Holy Quran. The authors used the original Quran corpus for training Doc2vec model, and QurSim [6] for the test. To predict if pairs of verses are semantically related, the cosine similarity was calculated among their associated vectors embedding.

The work described in [10] explored Word2vec model to find similar verses in the Holy Quran. However, the authors trained their models on seven English translations of the Holy Quran instead of its original Arabic version. Both word2vec models (i.e., Skip-Gram and CBOW) have been used to learn word embedding from Quran English translations. Then, the mean of word embedding constituting each verse has been taken to compute verse embedding. To find similar verses, the cosine similarity measure was computed among verses embedding.

In [11], the authors proposed a framework for semantic search using the Quran's ontology. The proposed framework includes the following six modules: Quranic Ontology, Quranic Database, Natural Language Analyzer, Semantic Search Model, Keyword Search Model, and Scoring and Ranking Model. However, no evaluation results were performed to demonstrate its effectiveness.

Other efforts have been made to develop semantic search tools for the Holy Quran's translated versions. For instance, in [12], the authors present a framework for both concept- and keyword-based English search of the Holy Quran. To implement their concept-based search tool called Qur'an enhanced search tool, the author first created a Quranic English WordNet database (QEWN) based on Princeton WordNet and enriched it with new terms from English Quran translations. This resource was used in query expansion. Furthermore, they developed a vocabulary of Quranic concepts in the form of a conceptual hierarchy using automatic term recognition techniques. The evaluation results of the proposed tool are encouraging, with an average recall and precision of 58.8% and 59%, respectively.

Recently, a concept-based search tool for the Holy Quran (QSST) was proposed [3]. The authors used CBOW model to learn word representation. Then, features' vectors of both Quranic topics and the input query are computed. To retrieve the most relevant verses for the user query, the cosine similarity between topics and query vectors was computed. The performance of QSST is found to be encouraging; the average precision, recall, and F-score are 76.91%, 72.23%, and 69.28%, respectively.

When analyzing related studies on semantic search in the Holy Quran literature, we notice that most researchers present a search by ontology rather than a semantic search approach. In addition, most of these studies focus on concept-based search using Quran ontology. There is a lack of tools that enable users to search by question. Furthermore, the majority of researches neglect the contextual meaning of the words and their semantic relations. The main goal of this study is to overcome these limitations.

3. PROPOSED APPROACH

In this paper, we propose a new semantic search approach for the Holy Quran. The proposed approach seeks to find all semantically related verses to the user input query. It consists of three modules, as shown in Figure 1. The first module concerns the Quran discourse segmentation. The second module concerns the search for the best word vector representation model for the Holy Quran. The third module concerns the semantic retrieval

model. We will discuss each module in the following subsections.

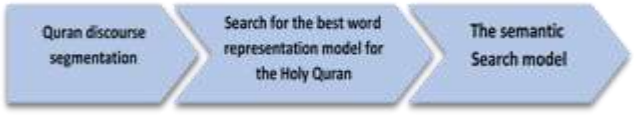

Figure 1. Proposed approach modules

A. Module 1: Quran Discourse Segmentation

Discourse segmentation is a central step in discourse analysis. It consists of segmenting an input text into smaller text units, called discourse units, based on the presence of predefined discourse markers [13]. Segmenting the Quranic text into discourse units is the key component of our approach. It allows us to separate the covered subjects within each verse, which enables a better representation of their meanings. To this end, we propose a new discourse segmentation method based mainly on the presence of particular marks used in Quran recitation. The proposed method consists of:

- Firstly, segment the Quran text into verses.
- Then, split each verse into discourse units based on specific Quran recitations marks (Tajweed marks). Recitation marks are punctuation marks in the shape of abbreviated signs used in the Holy Quran to correctly read and understand the Quran.

In our study, our focus is solely on recitation stop marks, specifically '**the compulsory stop marks'** {'مۡ'} and the '**Permissible Stop marks** {'صلے', 'قلے', 'ج', 'ص'}. These symbols are employed to denote the point at which the reader of the Quran should halt, as the intended meaning of the passage has been conveyed.
Relying on these marks to segment each verse into discourse units ensures the coherence of the generated discourse units and preserves their intended meaning as well as the meaning of the verse as a whole.

Depending on the presence of the above-mentioned stop marks, a verse may be segmented into many discourse units (cf. example 3), or it may be used as a single discourse unit. In the following example, the verse is segmented into three discourse units (between '[]').

(3) [وَإِذۡ قَالَ رَبُّكَ لِلۡمَلَٰٓئِكَةِ إِنِّي جَاعِلٌ فِي ٱلۡأَرۡضِ خَلِيفَةٗۖ **1**] [قَالُوٓاْ أَتَجۡعَلُ فِيهَا مَن يُفۡسِدُ فِيهَا وَيَسۡفِكُ ٱلدِّمَآءَ وَنَحۡنُ نُسَبِّحُ بِحَمۡدِكَ وَنُقَدِّسُ لَكَۖ] **2**قَالَ إِنِّيٓ أَعۡلَمُ مَا لَا تَعۡلَمُونَ[**3**

English translation: **[***When your Lord said to the angels, "I am placing a successor on earth.***]1 [***They said, "Will You place in it someone who will cause corruption in it and shed blood, while we declare Your praises and sanctify You***]2 [***He said, I know what you do not know.***]3**

Segmenting the Quran into discourse units generated a new dataset that we called Quran discourse units dataset.

B. Module 2: Search for the Best Word Representation Model for the Holy Quran

Word vector representation, also called word embedding, is one of the most effective techniques widely used in NLP for encoding word meaning in a low-dimensional space [14]. These models represent each word as a dense vector of real values that capture its underlying semantic and syntactic properties. Word embedding can be classified into two categories: contextualized and non-contextualized models. Non-contextualized word representation models such as Word2Vec [15], GloVe [16], and FastText [17] generate a single representation for each word, regardless of its context. While contextualized models move beyond word-level semantics. These models associate for each word multiple representations according to the used context. ELMo [18] and FLAIR [19] are among the most powerful contextualized embedding models.

The selection of the appropriate model that can accurately capture the inherent meaning of words in the Holy Quran was a serious problem, as it can significantly affect the performance of our approach. To address this issue, we chose to investigate the performance of the aforementioned five models on assessing semantic relatedness in the Holy Quran at verse level.

We focused on verse level rather than word level because our system should detect and provide semantically related verses rather than related words to the user's query. This latter can be a question, a verse, or a set of words. The followed methodology consists of five key phases, as shown in Figure 2. We will go over each phase in detail in the following subsections.

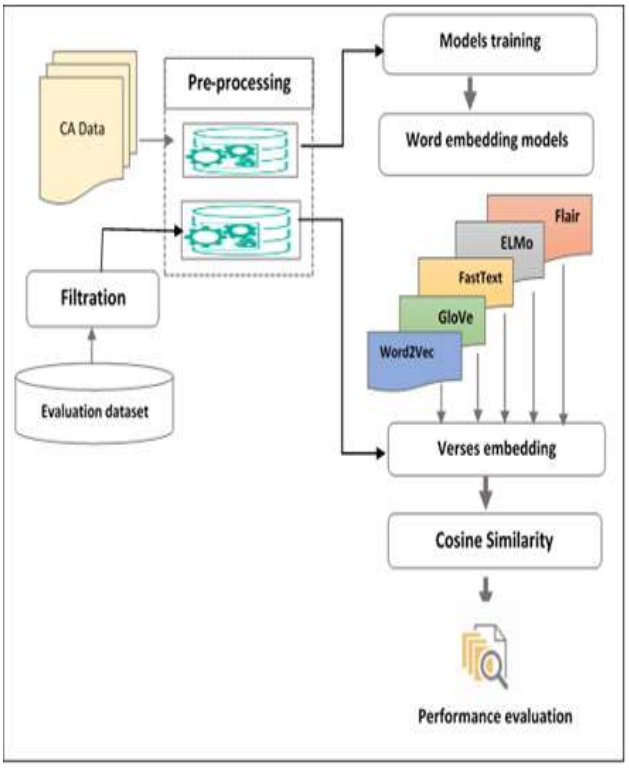

Figure 2. Main phases in word embedding models investigation

1) CA Data Collection

The present research focuses on the original Arabic version of the Holy Quran, not its translations. Therefore, we initially selected the original Arabic Quran dataset [20] to train our models. The dataset is a CSV file that contains three columns:

- **Surah ID:** the number of chapters. It ranges from 1 to 114 (114 is the total number of chapters in the Holy Quran).
- **Verse ID:** the verse number. It ranges from 1 to 6236.
- **Verse text:** the verse written in Arabic with diacritics.

Since the training of our models requires a large dataset, which is not the case for the Quran dataset, it was necessary to increase the size of our data. To this end, we built a classical Arabic corpus from two classical Arabic resources: the King Saud University Corpus of CA (KSUCCA) [21] and the Watan-2004 corpus. The KSUCCA corpus consists of 46 million words. It contains CA texts covering the following categories: religion, literature, sociology, linguistics, science, and biography. The Watan-2004 corpus consists of 20,000 articles covering six different topics: religion, economy, sports, local news, culture, and international news. Table 1 summarizes the statistics of the used datasets.

TABLE I. CHARACTERISTICS OF THE USED DATASET

Corpus/dataset	Covered Topics	Number of Words
The Holy Quran dataset [20]	114 chapters and 6236 verses	78245
KSUCCA [21]	religion, literature, sociology, linguistics, science, and biography	50602412
The Watan-2004 corpus	religion, economy, sports, local news, culture, and international news	106000000
Total number of words in the training dataset		156 680 657

2) Text Pre-Processing

Pre-processing aims to reduce inconsistency and word ambiguity for better word representation [22]. This step consists of five main tasks: diacritics removal, tokenization, normalization, linking words removal, and stemming.

- **Diacritics Removal:** This step consists of removing diacritical marks, which are added above or below a word, such as ‘ـَ’, ‘ـِ’, ‘ـُ’, ‘ـً’, ‘ـّ’, and ‘ـْ’.
- **Tokenization:** it concerns text cleaning by removing punctuation marks, numbers, and special characters, as well as splitting the text into separate words called tokens.
- **Normalization:** In Arabic, characters can have different variations due to the use of dots. This property can negatively affect both word representation and sentence similarity calculation. Normalization is the process of unifying the different forms of the same character to eliminate variations and make the text more consistent. The normalization of characters is performed using the following rules:
  - Replace the letter ‘أ’ with ‘ا’.
  - Replace the letters ‘ة’ and ‘ـة’ with ‘ه’ and ‘ـه’, respectively.
  - Remove the elongation: e.g., the word ‘العالميـــن’ is replaced with ‘العالمين’.
- **Linking-Words Removal:** linking words are conjunctions, pronouns, and prepositions. These words perform a syntactic function but do not indicate a subject or a significant meaning. In the case of the Holy Quran, we can't name these words as stop words or non-informative words due to the book's sacredness. Here, and after an in-depth analysis of the Holy Quran, we have compiled a list containing about 170 linking words.

- **Stemming:** Stemming is the process of reducing inflected words to their canonical form (stem), by removing affixes attached to them [22]. For instance, words like '**استخراج**', '**خروج**', and '**أخرج**' are reduced to one stem '**خرج**'. For the Arabic language, there are two dominant stemming approaches, namely light-based stemming (known as affixes removal) and root-based stemming, which relies on linguistic morphological analysis to extract word roots [13]. Following a comparative study between ten (10) stemming algorithms regarding Arabic text similarity at the sentence level [23], it has been shown that the best results were achieved using Farasa stemmer [24], and ARLSTem [25]. This is why we chose to use Farasa stemmer in our research. It's to be noted that the same pre-processing steps were carried out on the used evaluation dataset.

3) Building Word Embedding Models

In this phase, the chosen word embedding models are trained on the pre-processed large dataset. Both models of Word2Vec (i.e., CBOW and Skip-Gram) were used in this investigation. To build each model, we ran the training process several times to tune its optimal hyper-parameters. In the end, we have built six-word embedding models to be used as inputs for the next phase.

4) Verses Embedding

The trained models generate a dense vector representation for each word. However, sentence embedding is required in our study as we focus on verse level. Several studies have proven the effectiveness of using averaged word embedding to compute sentence embedding [26, 27]. This is why, in our research, we have calculated verse embedding as the average of their words embedding. For Flair model, we have applied '*Document Pool Embeddings*' method to compute verses embedding. Consequently, for each model, an embedding vector was computed for each verse in the evaluation dataset.

5) Measuring Similarity

Cosine similarity [7] is the most widely used metric in word embedding models. It defines the cosine of the angle between two vectors, which can be calculated as a normalized dot product of the two vectors, as shown in Eq 1. A cosine similarity close to '1' indicates that the two vectors have the same direction and their corresponding sentences are strongly related.

$$Sim(V_1, V_2) = \cos\theta = \frac{V_1.V_2}{\|V_1\| \times \|V_2\|} \quad (1)$$

Each pair of verses in the evaluation dataset was converted into two semantic vectors. We used cosine similarity to compute the similarity between these vectors, and thus estimate the degree of relatedness between their corresponding verses.

Once the performance evaluation of each model is performed, the best model is chosen to be used as input for the third module in our research that concerns the semantic search model.

C. Module 3: The Semantic Search Model

One of the key ideas of this research is the discursive segmentation of Quranic verses in order to properly separate their discussed topics. Consequently, instead of comparing the user's input query against the whole verse, it is compared against its constituent discourse units. A discourse unit can be a clause or a sentence that discusses a single subject and expresses a single meaning. Furthermore, we use the best word vector representation model to map Quranic discourse units and the user's input query to semantic vectors that capture their underlying meaning. Figure 3 presents the main steps of the semantic search model. These steps are discussed in detail in the following sub-sections.

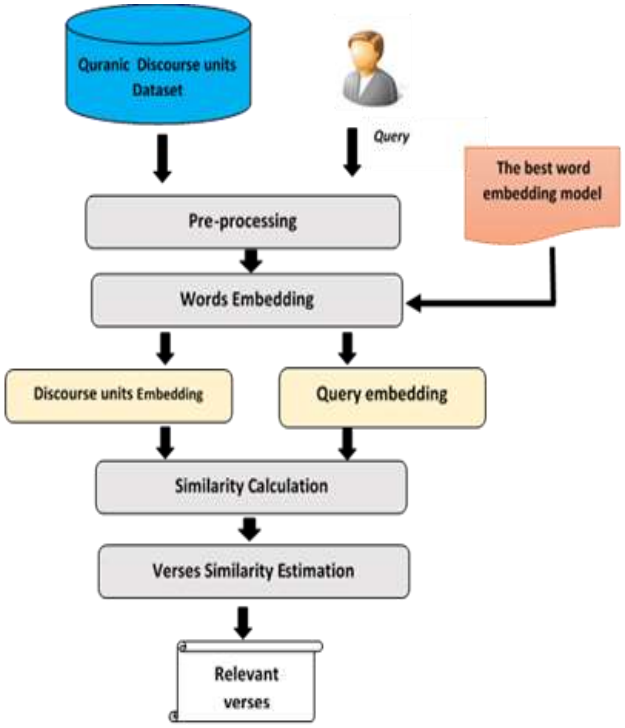

Figure 3. The semantic search model

1) Pre-processing

Pre-processing is a common step for both the user's query and the Quranic discourse units dataset. It involves diacritics removal, tokenization, normalization, linking words removal, and stemming, as described in Section 3.2. Figure 4 presents an output discourse unit after applying the pre-processing step.

Figure 4. An example showing the result of the pre-processing.

2) Words Embedding

Once the performance evaluation of the investigated word embedding models is achieved, the best model is selected to create a semantic vector representation for each word in the pre-processed discourse unit's dataset. Then, an embedding vector is computed for each discourse unit. As a result, for each verse in the Holy Quran, one or several embedding vectors have been associated, depending on the number of its constituent discourse units. In the same way, an embedding vector is created for the user's input query.

3) Similarity Calculation

We used the cosine similarity measure (described in Section 3.2.5) to estimate the similarity among the query and all discourse units embedding vectors in our dataset. The computed similarities are then assigned to discourse units as scores that reflect their relevance to the user input query. A score close to 1 means that the corresponding discourse unit is strongly related to the user input query.

4) Verses Similarity Estimation

A semantically related verse is a verse that includes at least one discourse unit that discusses the subject of the user's query. This means that it is not necessary that all discourse units constituting the verse be semantically related to infer that the whole verse is relevant to the user query. This is why, in our research, we choose to select for each verse in our dataset the highest similarity score of their constituent discourse units. The selected score is assigned to each verse to represent its semantic relatedness to the user's query. This way, even long verses that cover several topics and briefly refer to the subject of the user query will be well-ranked and considered.

4. EXPERIMENTS AND RESULTS

In this section, we first present the evaluation results of the second module, which concerns the investigation of word vector representation models in Section 4.1. Then, the evaluation results of the proposed approach are presented in Section 4.2.

A. Evaluation Results of Module 2: Search of the Best Word Representation Model

1) Evaluation Dataset

To evaluate the performance of our models in assessing semantic relatedness at the verse level, we have used the QurSim dataset [6], as it is the only resource available for this task. The QurSim dataset is a CSV file that contains 7679 pairs of verses labeled with three labels, as follows:

- 2: strongly related.
- 1: related.
- 0: non-related.

To be able to use the dataset, it was necessary to perform some interesting tasks, including mapping the dataset to text data, redundancy elimination, and filtration.

- ***Mapping the dataset to text data*:** The dataset contains numeric values corresponding to the number of chapters and verses. However, in our work, we need verses as input. Therefore, we have used the Quran dataset [20] to map all the numerical values to their corresponding verses.

- ***Redundancy Elimination*:** When checking the dataset, we noticed about 600 records of duplicated pairs of verses and more than 170 records of duplicated pairs annotated with two different labels. To ensure the consistency of our dataset, we have removed all redundant pairs and duplicated pairs labeled differently.

- ***Dataset Filtration***: In QurSim, many non-related verses are labeled as strongly related or related, and vice versa (see Table 2). This is because these labels were not annotated and checked by professional human experts. The presence of such records in the dataset will negatively affect the obtained results. To overcome this problem, we first selected only the most accurate annotated verse pairs in QurSim. The selected pairs were then attentively checked by five qualified human experts. As a result, a new dataset of 750 pairs of verses was

created to evaluate the investigated models. Henceforth, this dataset is referred to as Quranic-related verses (QURV).

TABLE II. EXAMPLES OF NON-RELATED VERSES LABELED AS STRONGLY RELATED IN QURSIM.

Verse 1	Verse 2	Label
"لم يلد ولم يولد" "*He neither begat nor was begotten*."	"و جعلوا بينه وبين الجنة نسبا ولقد علمت الجنة إنهم لمحضرون" "*And they have set up a kinship between Him and the jinn, while the jinn certainly knows they will indeed be presented*."	2
"كما أنزلنا على المقتسمين" "*Even as We sent down on the dividers (or swearers)*."	"أهؤلاء الذين أقسمتم لا ينالهم الله برحمة ادخلوا الجنة لا خوف عليكم ولا أنتم تحزنون" "*Are these the ones concerning whom you swore that Allah will not extend them any mercy? Enter the paradise; you shall have no fear, nor shall you grieve*."	2

2) Evaluation metrics

To evaluate our embedding models' performance, we used Spearman correlation coefficients. This metric estimates the effectiveness of text similarity models by quantifying how well their scores align with human similarity scores. The Spearman correlation coefficient is computed using Eq 2.

$$p = 1 - \frac{6\sum_{i=1}^{n}(x_i - t_i)^2}{n(n^2-1)} \qquad (2)$$

Where:

n : The number of verses' pairs.
x_i : The i^{th} human gold standard.
t_i : The i^{th} text similarity method score

Spearman correlation values range from -1 to +1. A value close to '1' means a high relationship between the model and human scores, proving the effectiveness of the model. -1 indicates a perfect inverse relationship, and 0 indicates no relationship.

3) Experimentation

In our experiments, we first performed some basic tasks. For instance, to create the vocabulary of each model, words occurring less than 3 times in the training dataset were removed. Regarding models training, several hyper-parameters were explored to select the best configuration for non-contextualized models. However, for ELMo model, we choose to use the best hyper-parameters configuration already explored in [28] for Arabic to minimize the time required in testing other hyper-parameters configurations from scratch. For Flair model, we used an LSTM with 512 hidden states and one layer. The used hyper-parameters for each model are given in Table 3.

TABLE III. TRAINING HYPER-PARAMETERS

Model	Hyper-parameters
CBOW	Window size = 10, minimum word count = 3, and the embedding dimension = 200
Skip-Gram	Window size = 10, minimum word count = 3, and embedding dimension =200
GloVe	Window size set to 10, embedding dimension= 150
FastText	window size = 7 , Vector size = 200
ELMo	batch_size = 128, 'bidirectional': True,b'lstm': {'cell_clip': 3, 'dim': 4096, 'n_layers': 2, 'proj_clip': 3, 'projection_dim': 512,'use_skip_connections': True}, 'char_cnn': {'activation': 'relu', 'embedding': {'dim': 16}, 'max_characters_per_token': 20,'n_highway': 2}
Flair	hidden_size=512, nlayers=1, max epochs = 20, sequence length= 12, mini-atch_size=16.

We used Python 3.8.0 to implement the three modules of the proposed approach with several tools and libraries. For instance, for data pre-processing, we used Pyarabic and Farasa toolkits. To build Word2Vec and FastText models, we used the implementations provided by Gensim Python library. We also used Flair platform[1] and "tensorflow_gpu-2.3.0" to build Flair and ELMo models, respectively. All experiments were performed using Google Colaboratory platform.

Our experiments have been carried out on a Dell machine with Windows 10 as operating system and the following hardware setup: Intel(R) processor Core(TM) i7-8750H CPU @ 2.20GHz, 2208 MHz, 6 Core(s), 12 Logical Processor(s) with RAM 8.00 Go, and a graphic card NVIDIA GeForce GTX 1060 with Max-Q Design.

[1] https://github.com/flairNLP/flair

4) Results and Discussion

Several experiments have been conducted to investigate the performance of the used models. In the first set of experiments, all models were trained on the pre-processed training dataset stemmed using Farasa stemmer [24]. Then, the Spearman correlation between the similarity scores calculated for each model and the human similarity scores in the evaluation dataset was computed. Figure 5 shows the results of this experiment. In the second set of experiments, we studied the impact of stemming on the performance of our models. To this end, we repeated the same set of experiments but without performing stemming. Figure 6 depicts the results of these experiments.

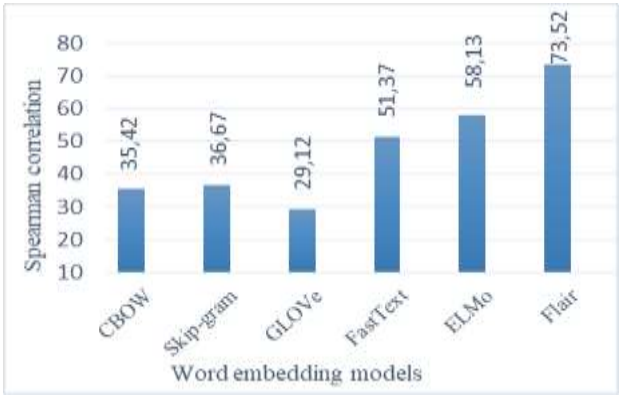

Figure 5. Models performance (with stemming)

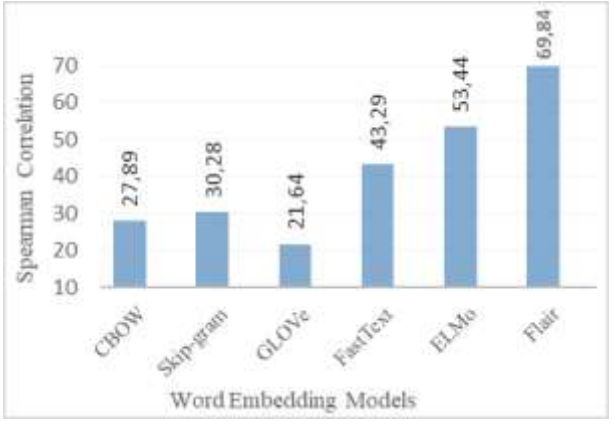

Figure 6. Models performance (without stemming)

When analyzing the obtained results, we notice that Flair model achieved the best performance with a score of 73.52% in terms of Spearman correlation. This indicates that Flair model trained on the Holy Quran and CA dataset can capture the semantics of words more accurately than the remaining non-contextual word embedding models. On the other hand, we observe that the worst performance was achieved by GloVe model with a score of 29.12%. We can also note that FastText model outperforms both Word2Vec and GloVe models with a score of 51.37%, which is relatively close to the performance of ELMo model. It is vital to stress that the performance achieved by the contextualized Flair model is encouraging when considering the rich semantics of the Holy Quran. Indeed, verses in the Holy Quran are semantically richer and more complex than those in CA and Modern Standard Arabic text.

However, without stemming, we can notice that all model's performance decreased. This degradation was approximately of 7.53%, 6.39%, and 6.48% in Spearman correlation scores for the CBOW, Skip-Gram and GloVe models, respectively. However, for Flair model, we notice a trivial degradation of 3.68% in Spearman correlation score. One possible explanation of this observation is due to the architecture of Flair model, which allows it to better learn sub-word representation, including stems and affixes. This allows it to project words correctly, regardless of their morphological variations.

Based on the achieved results, the Flair model already trained on the pre-processed dataset is selected to be used as the principal input of the third module of our semantic search system.

B. Evaluation Results of the Proposed Semantic Search Approach

We carried out three experiments to evaluate the performance of the presented semantic search approach. In the first experiment, a set of query are set as input and the achieved results for each query are compared against our gold standard in order to compute the precision, recall and F-score. In the second experiment, we examined the impact of discourse segmentation process. In the third experiment, the performance of the proposed approach in terms of precision is compared against the latest semantic search tool, (QSST) [3], already described in related work section.

1) Evaluation Metrics

The performance of the proposed approach is evaluated using precision, recall, and F-score. These metrics are defined as follows:

Precision: specifies the exactitude of results. In our case, it is defined as the number of relevant verses divided by the total

number of retrieved verses, as shown in Eq. 3.

$$precision = \frac{relevant\ verses\ retrieved}{total\ number\ of\ retrieved\ verses} \quad (3)$$

Recall: measures the coverage of results. It is calculated using Eq. 4.

$$recall = \frac{relevant\ verses\ retrieved}{total\ number\ of\ relevant\ verses} \quad (4)$$

F-score: the harmonic mean of recall and precision. It's calculated using Eq. 5

$$F - score = \frac{2 * precision * recall}{precision + recall} \quad (5)$$

2) Gold Standard

Since there is no available gold standard that can be used for semantic search in the Holy Quran, it was necessary to create our evaluation dataset. To this end, we chose to exploit our (QURV) dataset described in Section 4.1.1 to create our gold standard. First, we selected from the QURV dataset only the most strongly related verse pairs (annotated with label 2). Then, we manually annotated each pair of verses with their covered topic. As a result, we built a new dataset of semantically related verses covering seven different topics: monotheism, resurrection, righteousness of parents, astronomy, charity, Isa ibn Maryam, and prayer. After that, we carried out a manual search with the help of five qualified Islamic experts to enrich each topic class with its semantically related verses. Note that a verse may fall into multiple classes if it encompasses multiple topics. Finally, we built our gold standard, which consists of 290 verses covering seven different topics, as shown in Table 4.

TABLE IV. GOLD STANDARD CHARACTERISTICS

Topic	Number of verses
Monotheism/التوحيد	35
Resurrection/القيامة	150
Righteousness of parents/ بر الوالدين	8
Astronomy / الفلك	21
Charity/ الصدقة	31
Isa ibn Maryam/ عيسى ابن مريم	25
Prayer / الصلاة	20
Total	**290**

3) Results and Discussion

The proposed approach is evaluated using a set of queries collected from Islamic websites. For each input query, the retrieved verses are recorded and compared against our gold standard. Then, evaluation metrics are computed. Table 5 presents the performance of the proposed approach in terms of precision, recall, and F-score.

TABLE V. PERFORMANCE OF THE PROPOSED SEARCH APPROACH

Input query	Precision	Recall	F-score
هل هناك اله واحد Is there one single God?	93.33	80%	86.15%
كيف ستكون القيامة How will the resurrection be?	89.58 %	28.66%	43.42%
هل يجب الاحسان الى الوالدين Is it necessary to be kind to parents?	88.9%	100%	94.11%
خلق السماوات و الأرض Creation of the heavens and the earth	89.47%	80.95%	85%
الحث على انفاق المال Urging charity	96%	77, 41%	85.7%
الصلاة Prayer	78.26%	90%	83.7%
قصة عيسى ابن مريم The story of Jesus, son of Mary	100%	100%	100%
Average	**90.79%**	**79.57%**	**82.58%**

As shown in Table 5, the proposed system achieves promising results for most topics, notably 'Isa ibn Maryam'. The system can retrieve its semantically related verse with an F-score of 100%. For the remaining topics, the system's performance in terms of F-score ranges from 83.7% (for the topic 'prayer') to 94.11% (for the topic' righteousness of parents'). However, the worst accuracy is 43.42% for the topic "Resurrection". The system can retrieve only 43 of 150 relevant verses with a high precision of 89.58%. We can also note that the average precision of the proposed system is 90.79%; which proves that our system is quite good at retrieving semantically related verses and excluding irrelevant ones.

To explore the impact of discourse segmentation process, we repeated the first experiment, but using the original Quran dataset [20] instead of Quran discourse units dataset. Table 6 shows the result of this experiment.

From Table 6, we can notice that the overall system's performance has noticeably degraded by 29.68% and

30.07% in terms of average precision and recall, respectively. This means that the ability of the system to retrieve all semantically related verses to the input query has considerably decreased when considering verses as atomic units. However, the resurrection topic experiences a marginal degradation in precision and recall, estimated at 6.25% and 2%, respectively, compared to the other topics. This is due to the fact that most related verses to the resurrection topic were not subject to the discourse segmentation process because they are very short verses.

TABLE VI. SYSTEM'S PERFORMANCES WITHOUT USING DISCOURSE SEGMENTATION ES

Input query	Precision	Recall	F-score
هل هناك اله واحد Is there one single God?	59.27%	45.71%	51.61%
كيف ستكون القيامة How will the resurrection be?	83.33 %	26.66%	40.4%
هل يجب الاحسان الى الوالدين is it necessary to be kind to parents?	38.46%	62.5%	47.62%
خلق السماوات و الأرض Creation of the heavens and the earth	52%	61.9 %	56.52%
الحث على انفاق المال Urging charity	63.15 %	38.7%	48%
Prayer/ الصلاة	57.89 %	55%	56.41%
ما قصة عيسى ابن مريم	73.68%	56%	63.63%
Average	**61.11%**	**49.5%**	**52.03%**

In the third experiment, the performance of the proposed approach is compared against the most recent concept-based search tool, the Quranic Semantic Search Tool (QSST) [3]. QSST has already compared to other semantic search tools and it has proven its superiority. To evaluate the performance of QSST, a set of query concepts were set as input, then obtained results for each query were evaluated and the performance in terms of precision was computed. The following queries were used in the evaluation of QSST: 'قيام الليل/ Night Prayer', 'السيدة مريم/ Maryam', 'بر الوالدين/Righteousness of parents', 'علم الفلك/Astronomy'. In this experiment, we chose the same queries to test our approach. Figure 7 and Table 8 provide a performance comparison of the proposed approach against QSST.

Figure 7. Performance comparison in terms of precision of the proposed approach against QSST

TABLE VII. PERFORMANCE COMPARISON IN TERMS OF AVERAGE PRECISION OF THE PROPOSED APPROACH AGAINST QSST

Research work	Average precision
QSST	89.31%
Proposed approach	92.7%

As shown in Figure 7, Table 7, the proposed approach outperforms QSST in terms of precision. The average precision of our approach is estimated at 92.7%. Compared to the semantic richness of the Holy Quran and its linguistic peculiarities, such performances are very well received.

Overall, achieved results are very encouraging and prove the effectiveness of the proposed approach and the ability of the contextualized word representation Flair model to capture the contextual meanings of words and their semantic information, even in the Holy Quran.

5. CONCLUSION

In this paper, we propose a new semantic search approach for the Holy Quran. The proposed approach consists of three modules. The first module concerns the segmentation of verses into discourse units based on specific Quran recitation stop marks. The second module aims to identify the best word representation model for mapping the resulting discourse units into semantic vectors. To this end, we trained advanced word representation models on a classical Arabic corpus and investigated their performance in detecting semantic relatedness in the Holy Quran. Evaluation results show that Flair model achieves the best performance. Therefore, we chose this model as input for the third module to map the user's query and the generated discourse units into semantic vectors. Finally, we compute cosine similarity between the input query and all discourse unit vectors to retrieve relevant verses for the user's query.

The evaluation results are very promising and prove the strength of the proposed approach, especially the role of Quran discourse segmentation in improving the

accuracy of semantic search. In addition, these results demonstrate the power of the contextual embedding Flair model in capturing the contextual meaning of words and detecting the semantic relationships between verses in the Holy Quran.

In future work, we plan to investigate the use of recent transformer-based models to detect semantically related verses to the user's query. Moreover, we will devote efforts to extend this research work by developing an extensive evaluation dataset that can be used as a gold standard for semantic search in the Holy Quran. Doing so, will encourage researchers to explore this exciting field and compare their results against a unified gold standard.

REFERENCES

[1] D. Chandrasekaran and V. Mago, "Evolution of semantic similarity—a survey", ACM Computing Surveys (CSUR). vol.54. no. 2 , pp.1-37, 2021.

[2] S. Kim, N. Fiorini,W. J.Wilbur, and Z. Lu, "Bridging the gap: Incorporating a semantic similarity measure for effectively mapping pubmed queries to documents", Journal of biomedical informatics, vol. 75, pp.122– 127, November 2017.

[3] E. H. Mohamed, and E. M. Shokry, "QSST, A Quranic Semantic Search Tool based on word embedding". Journal of King Saud University-Computer and Information Sciences, vol 34. no.3, pp. 934-945, 2022.

[4] R. El-Deeb, A. M., Al-Zoghby and S. ELMougy, "Multi-corpus-based model for measuring the semantic relatedness in short texts (SRST)". Arabian Journal for Science and Engineering, vol.43, pp. 7933-7943, 2018.

[5] M. Alshammeri, E. Atwell, and M. ammar Alsalka, "Detecting Semantic-based Similarity Between Verses of The Quran with Doc2vec", Procedia Computer Science . vol. 189, pp.351-358,2021.

[6] A. B. Sharaf, and E. Atwell. QurSim: "A corpus for evaluation of relatedness in short texts ". In Proc of the Eighth International Conference on Language Resources and Evaluation (LREC'12), pp. 2295-2302, 2012.

[7] H. Schutze, C. D. Manning, & P. Raghavan. "*Introduction to information retrieval*". Cambridge University Press, 2008.

[8] A.-B.M Sharaf; Atwell, E., QurAna, "Corpus of the Quran annotated with Pronominal Anaphora". In: LREC 2012, pp. 130–137,2012.

[9] M. Akour, I. M. Alsmadi, & I. Alazzam, "MQVC: Measuring quranic verses similarity and sura classification using N-gram", WSEAS Transactions on Computers, vol. 135, pp. 485-491,2014.

[10] S. Saeed, S. Haider, and Q. Rajput, "On finding similar verses from the Holy Quran using word embeddings". In 2020 International Conference on Emerging Trends in Smart Technologies (ICETST), IEEE, pp. 1-6,2020

[11] Alqahtani, M. and Atwell, E., "Arabic Quranic Search Tool Based on Ontology", in Natural Language Processing and Information Systems, pp. 478–485, 2016.

[12] Afzal, H. and Mukhtar, T., "Semantically enhanced concept search of the Holy Quran: Qur'anic English WordNet." Arabian J. Sci. Eng. vol. 44 no.4, pp . 3953-3966, 2019.

[13] S. Lagrini, N. Azizi, and M. Redjimi, "Exploiting discourse relations to produce Arabic extracts". *International Journal of Reasoning-based Intelligent Systems*, vol. *14*(2-3), pp. 130-143, 2022.

[14] B.Chiu, and S. Baker, "Word embeddings for biomedical natural language processing: A survey". Language and Linguistics Compass, vol. 14, e12402,2020.

[15] T. Mikolov, K. Chen, G. Corrado, and J. Dean, "Efficient estimation of word representations in vector space", arXiv preprint arXiv:1301.3781, 2013.

[16] J. Pennington, R. Socher, and C. D. Manning, "Glove: Global vectors for word representation", in Pro of the 2014 conference on empirical methods in natural language processing (EMNLP), pp. 1532–1543,2014.

[17] P. Bojanowski, E. Grave, A. Joulin, and T. Mikolov. "Enriching word vectors with subword information. Transactions of the association for computational linguistics", vol.5, pp. 135-146, 2017.

[18] M.E., Peters, M. Neumann, M. Iyyer, M. Gardner, C. Clark, K. Lee, and L. Zettlemoyer., "Deep contextualized word representations ". arXiv preprint arXiv:1802.05365, 2018

[19] A., Akbik, D., Blythe, and R., Vollgraf, "Contextual string embeddings for sequence labeling". In *Proceedings of the 27th international conference on computational linguistics* ,pp. 1638-1649, August 2018.

[20] Tanzil Documents, [Online]. Available. (accessed 13.12.2023).

[21] M. Alrabia, E. Atwell, A. Al-Salman, and N. Alhelewh, "KSUCCA: a key to exploring arabic historical linguistics ". Int. J. Comput. Linguist, vol.5 no.2, 27,2014.

[22] S. Lagrini, , and M. Redjimi, " A New Approach for Arabic Text Summarization". In *Proceedings of the 4th International Conference on Natural Language and Speech Processing (ICNLSP 2021),* pp. 176-185, 2021.

[23] M. O. Alhawarat, H. Abdeljaber, and A. Hilal, "Effect of stemming on text similarity for Arabic language at sentence level ". PeerJ Computer Science, 7, e530,2021.

[24] A. Abdelali, K. Darwish, N. Durrani, and H. Mubarak, "Farasa: a fast and furious segmenter for arabic ", in 15th annual conference of the north american chapter of the association for computational linguistics: human language technologies, pp. 11–16, 2016.

[25] K. Abainia., S. Ouamour., and H. Sayoud, "A novel robust Arabic light stemmer". Journal of Experimental & Theoretical Artificial Intelligence, vol . 29, pp. 557–573,2017.

[26] M. Pagliardini, P. Gupta, and M. Jaggi, "Unsupervised learning of sentence embeddings using compositional n-gram features", arXiv preprint arXiv:1703.02507,2017

[27] S. Arora, Y. Liang, and T. Ma, "A simple but tough-to-beat baseline for sentence embeddings". International conference on learning representations, 2017.

[28] H. Al-Bataineh, W. Farhan, A. Mustafa, H. Seelawi, and H. T. Al-Natsheh . "Deep contextualized pairwise semantic similarity for Arabic language questions". In proc of 2019 IEEE 31st International Conference on Tools with Artificial Intelligence (ICTAI), IEEE pp. 1586-1591,2019.