International Journal of Computing and Digital Systems
2025, VOL. 17, NO. 1, 1–14

http://dx.doi.org/10.12785/ijcds/1571013400

A New Semantic Search Approach For The Holy Quran Based On Discourse Analysis And Advanced Word Representation Models

Samira Lagrini[1] **and Amina Debbah**[2]

[1]*Computer Science Department, LABGED Laboratory, Annaba, Algeria*
[2]*Computer Science Department, LRI Laboratory, Annaba, Algeria*

Received 28 March 2024, Revised 04 August 2024, Accepted 20 August 2024

Abstract: Semantic search is an information retrieval technique that aims to understand the contextual meaning of words to achieve 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 capacity. This paper presents a new semantic search approach for the Holy Quran. Our 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 key modules. The first module concerns the discourse segmentation of Quranic text into non-overlapping 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 is investigated. The third module implements the semantic search process. The proposed approach achieves promising results, with an average precision of 90.79% and a recall of 79.57%, showcasing the effectiveness 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, Semantic search, Artificial Intelligence, Holy Quran.

1. Introduction

Semantic relatedness is a Natural Language Processing (NLP) task that involves assessing the level of relatedness between two text units in a given language [1]. Often, semantic relatedness is used synonymously with semantic text similarity. However, semantic relatedness considers a broader perspective by analyzing the common semantic properties of two words, making it a more encompassing area that includes semantic similarity.

Assessing the semantic relatedness between text units plays an essential role in various NLP tasks, including information retrieval [2] and semantic search. Semantic search is an advanced 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]. Unlike lexical search, which focuses on the lexical matching of query words while ignoring their contextual meaning and semantic relationships, a semantic search system must be adept at interpreting the user's query. It should accurately find semantically related documents, even when a lexical match of query words is not present.

There is a persistent need for such a system for the Holy Quran to search semantically related verses that discuss the user's topic of interest. Most existing search systems rely on exact word matching, which often misses relevant verses that use different vocabulary to convey deeper contextual meanings related to the user's topic of interest. The rich semantics and unique style of the Holy Quran present significant challenges for semantic search systems, requiring more advanced techniques.

The objective of this paper is to address these challenges and surpass the limitations of current methods by presenting a novel semantic search approach specifically designed for the Holy Quran.

The Holy Quran is the most sacred religious book for more than 2 billion Muslims and their source of guidance. It is an exceptional Classical Arabic (CA) text with an

E-mail address: samira.lagrini@univ-annaba.dz, amina.debbah@univ-annaba.dz

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 analysis [5]. This is why Muslims and even non-Muslims strive to understand its content and deeply analyze its semantically related verses to get an accurate and broad explanation of the intended meanings. Studying all semantically related verses together provides an exhaustive understanding of the target subject, helping to gain in-depth religious knowledge and understand various judgments, thus offering deeper insights into the sacred text's teachings and themes.

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

(١) خُشَّعًا أَبْصَارُهُمْ يَخْرُجُونَ مِنَ الْأَجْدَاثِ كَأَنَّهُمْ جَرَادٌ مُنْتَشِرٌ (القمر، ٧)

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

(٢) مُهْطِعِينَ مُقْنِعِي رُءُوسِهِمْ لَا يَرْتَدُّ إِلَيْهِمْ طَرْفُهُمْ وَأَفْئِدَتُهُمْ هَوَاءٌ (ابراهيم ، ٤٣)

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.

(٣) [وَما مِن دابَّةٍ فِي الأَرضِ وَلا طائِرٍ يَطيرُ بِجَناحَيهِ إِلّا أُمَمٌ أَمثالُكُم]١ [ما فَرَّطنا فِي الكِتابِ مِن شَيءٍ]٢ [ثُمَّ إِلىٰ رَبِّهِم يُحشَرون]٣ (الانعام ٣٨)

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.

(٤) [هَذا كِتابُنا يَنطِقُ عَلَيكُم بِالحَقِّ]١ [إِنّا كُنّا نَستَنسِخُ مَا كُنتُم تَعمَلُونَ]٢ (الجاثية، ٢٩)

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 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 correctly targeting the covered topics in verses via our original discourse segmentation method. To the best of our knowledge, the present research is the first to address the area of Quran discourse analysis and combines related information with advanced techniques in word representation to improve Quran semantic search results.

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 the 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 rest of the paper is structured as follows: Section 2 provides a concise overview of recent research 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

While the Holy Quran has recently garnered some attention as a research subject in NLP, the number of published studies in this area is still limited. This scarcity is primarily due to the significant challenges of the Holy Quran's linguistic structure, its profound meanings and rich semantics. These complexities make it difficult to apply conventional NLP techniques effectively.

In this section, we provide an exhaustive review of the existing research on semantic relatedness and semantic search related to the Holy Quran over the last decade.

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

In [9], the authors used the 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. 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 verse's representation by the shared concepts. The second level used a list of synonyms collected using Arabic online dictionaries to enrich the verse's vector. The cosine similarity measure was used to compute the similarity among each pair of verses.

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] used 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 [12], [13], [14], [15]. For instance, in [15], a semantic search framework based on Quran WordNet is introduced to find the most relevant verse translations . The authors demonstrate that the use of Quran WordNet and a Part of Speech Tagger enhances the semantic search and yields promising results for sentiment analysis and concept-based search.

In the same context, the authors in [12] present a framework for concept and keyword-based English search for the Holy Quran. To implement their concept-based search tool, the authors 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.

In [3], a concept-based search tool for the Holy Quran (QSST) was proposed. The authors used CBOW model to learn word representation. Then, the features vectors of Quranic topics and the input query are computed. The cosine similarity between topics and query vectors was computed to retrieve the most relevant verses for the user query. 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.

Recently, in [16], the author examined the performance of large language model (LLM) embeddings for semantic search in the Holy Quran. The proposed methodology involved dividing the Quranic text into verses, then transforming these verses and the user's input query into embedding vectors. To this end, the GPT embedding model developed by Open AI was utilized. To retrieve similar verses to the user's query, the cosine similarity measure is computed between the query and verse embedding vectors. The evaluation results of the proposed methodology indicate that the GPT model embeddings outperform the three traditional embedding techniques used as baselines. However, no comparisons with existing related works were conducted. Additionally, there is a significant ambiguity regarding the evaluation dataset and the employed evaluation strategy.

An analysis of related studies on semantic search in Holy Quran literature reveals that most researchers focus on ontology-based search methods rather than true semantic search approaches. While ontology-based methods provide a structured framework for organizing and understanding the relationships between concepts within the Holy Quran, they have significant drawbacks. These methods are often time-consuming, requiring extensive manual effort, and they fall short of capturing the deep contextual dependencies inherent in the Holy Quran.

Conversely, traditional information retrieval techniques, such as TF-IDF and cosine similarity, are effective for lexical search but lack the ability to capture the contextual meanings and underlying semantics of Quran's words. This limitation is partially addressed by the use of non-contextualized word embedding in recent studies. While these embeddings offer some improvements in capturing word meanings, they still fall short of fully representing the rich and nuanced semantics of the Holy Quran.

In contrast, our approach diverges significantly from traditional methods. By combining the power of advanced contextualized word representation models with our Quran discourse segmentation method, we address the limitations of earlier studies and establish a more robust framework for semantic search in the Holy Quran. The following section details our proposed approach.

3. Proposed Approach

This paper introduces a novel semantic search approach for the Holy Quran. The proposed approach aims to identify all semantically related verses corresponding to the user's query. To achieve this, we developed three modules, as shown in Figure 1. The first module focuses on the discourse segmentation of the Quranic text. Each verse is segmented into non-overlapping discourse units based on predefined discourse markers. The second module evaluates the performance of several word representation models in capturing semantic information from the Holy Quran. The goal is to select the most appropriate model for transforming Quranic discourse units into semantic vectors. The third module addresses the semantic search process. Each module will be discussed in detail in the following subsections.

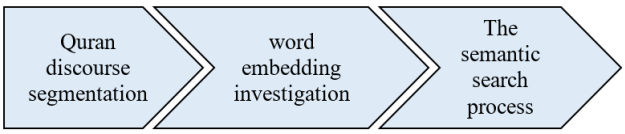

Figure 1. Proposed Approach Modules

A. Module 1: Quran Discourse Segmentation

Discourse segmentation is crucial in discourse analysis. It involves dividing an input text into smaller units, called discourse units, based on predefined markers [17]. Segmenting the Quranic text into discourse units is key to our approach, as it allows us to separate the subjects covered within each verse, facilitating better representation of their meanings. To this end, we propose a new discourse segmentation method based primarily on specific marks used in Quran recitation. The proposed method includes the following steps:

- Segmenting the Quran text into verses.
- Splitting each verse into discourse units based on specific Quran recitation marks (Tajweed marks). These marks are punctuation symbols that guide correct reading and understanding of the Quran.

In our study, we focus solely on recitation stop marks, specifically the following:

- The compulsory stop marks: 'م'
- The Permissible Stop marks: 'صلي','قلي', 'ص','ج'

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. Algorithm 1 presents the pseudocode of Quran discourse segmentation process. The algorithm takes as input the Quran dataset segmented into verses and the list of recitation stop marks, and provides as output a new dataset that contains both Quranic verses and their generated discourse units.

Depending on the presence of the above-mentioned stop marks, a verse may be segmented into many discourse units (cf. example 5), or it may be used as a single discourse unit (cf. example 6).

(٥) [أَلَيسَ اللَّهُ بِكافٍ عَبدَهُ]١ [وَيُخَوِّفونَكَ بِالَّذينَ مِن دونِهِ]٢ [وَمَن يُضلِلِ اللَّهُ فَما لَهُ مِن هادٍ]٣

Algorithm 1: Quran Discourse Segmentation

Input: The Quran dataset (QV)
T = {ص, ج, قلي, صلي, م}
Output: Quran discourse units dataset (QDU)

// (QV) dataset is a CSV file that contains
the Quran text already segmented into verses.

for (verse V_i in (QV)) **do**
{
S = {}
//The set of generated discourse units of each verse
DUs = V_i.split (ص، ج، قلي، صلي، م)
// Segment the verse into discourse units
based on the above marks

S.add(DUs)
}
if ($S \neq$ {}) **then**
Insert S to (QDU)
else
Insert V_i to (QDU)
end if
}
end for

Return (QDU)

English translation: [Does not Allah suffice His servant?]1 [They would frighten you of others than Him]2 [Yet whomever Allah leads astray, has no guide]3

(٦) [وَلَقَدْ خَلَقْنَا الْإِنسَانَ مِن سُلَالَةٍ مِّن طِينٍ]١

English translation: [Certainly We created man from an extract of clay]

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

Word vector representation, or word embedding, is a widely used NLP technique for encoding word meanings in a low-dimensional space[18]. These models represent each word as a dense vector of real values, capturing its semantic and syntactic properties. Word embedding can be classified into two categories: contextualized and non-contextualized models. Non-contextualized models, such as Word2Vec [19], GloVe [20] , and FastText [21] generate a single representation for each word, regardless of context. In contrast, contextualized models, like ELMo[22] and Flair[23] provide multiple representations for each word depending on its context.

Selecting an appropriate model that accurately captures the inherent meanings of words in the Holy Quran was a significant challenge, as it can significantly impact the performance of our approach. To address this issue, we investigated the performance of five models in assessing semantic relatedness in the Holy Quran at verse level. To the best of our knowledge, this is the first research to tackle this problem.

We mainly 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.

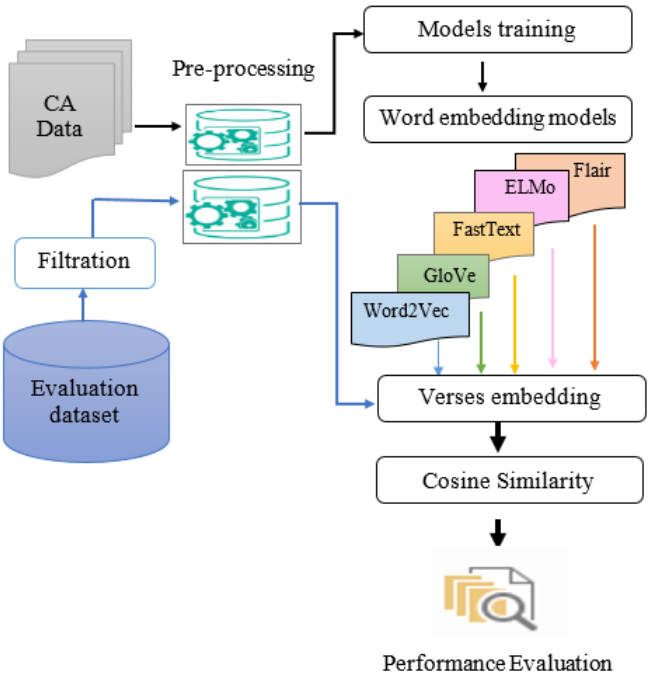

Figure 2. Main Phases in Word Embedding Models Investigation

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 [24] 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) [25] 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 I summarizes the characteristics of the used datasets.

TABLE I. Characteristics of the used dataset

Corpus/dataset	Covered Topics	Word count
The Holy Quran dataset [24]	114 chapters and 6236 verses	78,245
KSUCCA [25]	Religion, literature, sociology ,linguistics, science, biography	50,602,412
Watan-2004 corpus	Religion, economy, sports, local news, culture,international news	106,000,000
Training dataset word count		156,680,657

2) Text Pre-Processing:

Pre-processing aims to reduce inconsistency and word ambiguity for better word representation [26]. 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 words in the Holly Quran.
- **Tokenization**: It involves two primary activities: text cleaning and splitting. Text cleaning includes removing punctuation marks, numbers, and special characters, while splitting involves breaking down the cleaned text into separate words called tokens.
- **Normalization**: In Arabic, characters can exhibit various forms due to the presence of dots, which can negatively impact word representation and sentence similarity calculations. Normalization is the process of unifying the different forms of the same character to eliminate these variations and enhance text consistency. In our research, normalization is achieved through the following rules:
  - 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 [26]. For instance, words like 'خروج' 'استخراج' 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 [17]. Following a comparative study between ten (10) stemming algorithms regarding Arabic text similarity at the sentence level [27], it has been shown that the best results were achieved using Farasa stemmer[28], and ARLSTem [29]. This is why we chose to use Farasa stemmer in our research. It is important to note that the same preprocessing steps were applied to the 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 [30], [31]. 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 process.

C. Module 3: The Semantic Search Process

One of the key ideas of this research is the discursive segmentation of Quranic verses to properly separate their topics. Consequently, instead of comparing the user's input query to the whole verse, it is compared to its constituent discourse units. A discourse unit can be a clause or a sentence that discusses a single subject and conveys a single meaning. These discourse units are transformed into semantic vectors using the optimal word vector representation model, selected through an extensive investigation in Module 2. Figure 3 illustrates the main steps of the proposed semantic search model, which are discussed in detail in the following subsections.

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 Subsection B of Section 3. Figure 4 presents an output discourse unit after applying the pre-processing step.

2) Words Embedding

After evaluating the performance of the investigated word embedding models, the best model is selected to create a semantic vector representation for each word in the pre-processed discourse unit's dataset. Subsequently, an

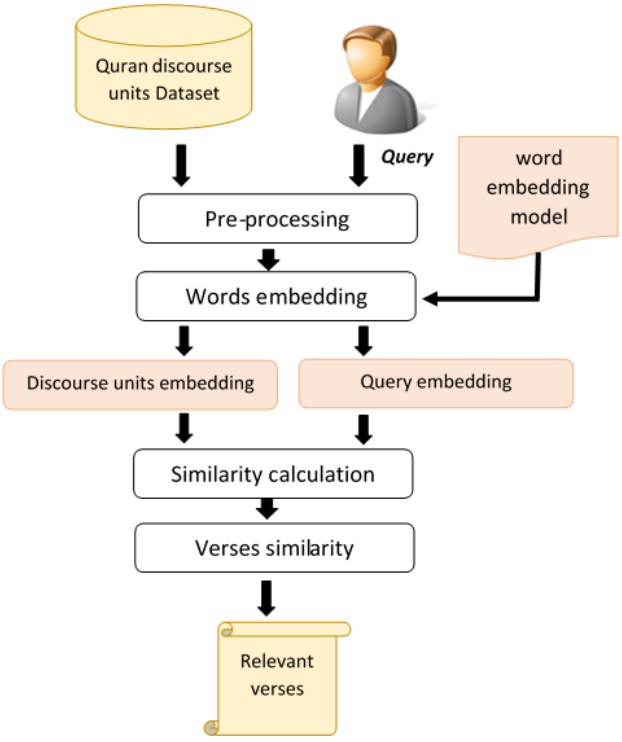

Figure 3. The Semantic Search Process

Figure 4. An Example Showing the Pre-processing Result.

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 already described in Subsection B. to estimate the similarity between 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 addressing the subject of the user's query. This means that it is not necessary for all discourse units within a verse to be semantically related to the query to infer that the entire verse is relevant. Therefore, in our research, we select the highest similarity score from the constituent discourse units for each verse in our dataset. This score is then assigned to the entire verse to represent its semantic relatedness to the user's query. This approach ensures that even long verses covering multiple topics and briefly mentioning the subject of the user's query will be well-ranked and considered.

Verses are then ranked in descending order based on their similarity scores. Top-ranked verses above a predefined threshold are selected as the system's output. Algorithm 2 presents the pseudocode for the semantic search process.

Having established our methodology, we evaluated its effectiveness through a series of experiments, detailed in the following section.

Algorithm 2: Quran Semantic Search

Input
(QDU) Quran discourse unit dataset
(q) User's input query
Output
$R = \{V_1, V_2, \ldots, V_X\}$
// related verses to the User's input query (q)

$R = \{\}$
Preprocessing (QDU)

// Create an embedding vector for each discourse unit in the preprocessed dataset
for verse V_i **in** (QDU) **do**
{
for discourse unit DU **in** V_i **do**
DU_i-embedding = Flair-embedding (DU)
end for
}
end for

// Create an embedding vector for the preprocessed user's query
Q = Preprocessing (q)
X = Flair-embedding (Q)

// Compute similarity between the query embedding and all discourse unit embedding vectors of each verse

for verse V_i **in** (QDU) **do**
{
for each discourse unit DU_x **in** V_i **do**
Sim (DU_x) = cosine-similarity (X, DU_x-embedding)
end for

Score V_i = Maximum (Sim (DU_x))

// The score of each verse is the highest similarity score of their constituent discourse units

If score V_i > threshold R.add(V_i)
}
end for

Verses-ranking (R)

print (R)
// Display relevant verses

4. Experiments And Results

First, we present the evaluation results of the second module, focusing on the investigation of word vector representation models in Subsection A. Next, we present the evaluation results of the proposed approach in Subsection B.

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 verse level, we have used the QurSim dataset [6]. It is the only resource available for this task. QurSim 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 representing chapter and verse numbers. However, in our work, we need verses as input. Therefore, we have used the Quran dataset [20] to map all 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. This is because these labels were not anno-

tated 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 accurately 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 verse pairs was created for model evaluation. Henceforth, this dataset is referred to as Quranic-related verses (QURV).

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.

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

Where:

- n: The number of verses' pairs.
- x_i: The ith human gold standard.
- t_i: The ith 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 [32] 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 II.

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 and "tensorflowgpu-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.

TABLE II. Training hyper-parameters.

Model	Hyper-parameters
CBOW	Window size: 10, min word count:3 embedding dim: 200
Skip-Gram	Window size: 10, min word count: 3 embedding dim:200
GloVe	Window size:10, embedding dim: 150
FastText	window size: 7 , Vector size: 200
ELMo	batch size : 128, bidirectional: True blstm: cell clip: 3, dim: 4096, nlayers: 2 proj clip: 3, projection dim: 512, use skip connections: True, char cnn: activation: relu, embedding: dim:16 max characters per token: 20
Flair	hidden size:512, nlayers: 1 max epochs : 20, sequence length: 12

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 [28]. 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.

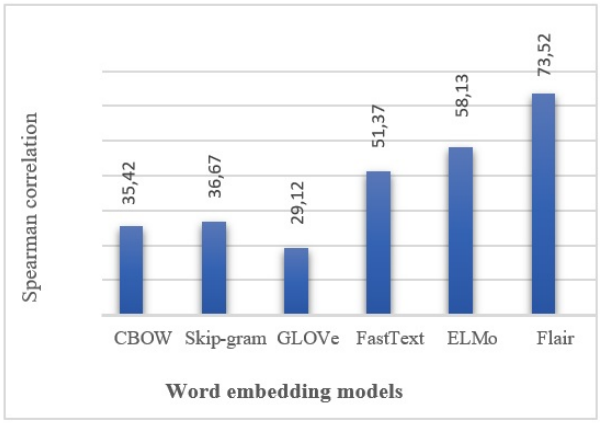

Figure 5. Models Performance (with Stemming)

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.

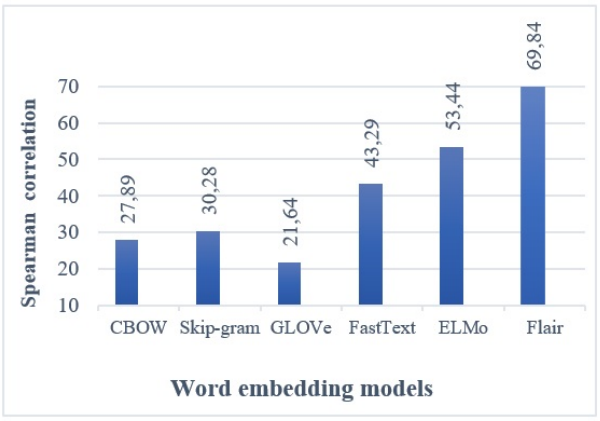

Figure 6. Models Performance (without Stemming)

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, sentences 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

To evaluate the performance of the proposed semantic search approach, we conducted two sets of experiments. In the first set, we used a predefined list of queries as input and compared the results against our gold standard to compute precision, recall, and F-score. In the second set, we assessed the impact of the discourse segmentation process on the performance of the semantic search system.

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 exactness of the obtained results in an information retrieval system. It measures the proportion of relevant documents retrieved among the total number of documents retrieved. In our context, it is defined as the number of relevant verses retrieved divided by the total number of retrieved verses, as shown in Eq. 3.

$$\text{Precision} = \frac{\text{relevant verses retrieved}}{\text{total number of retrieved verses}} \tag{3}$$

Recall: Measures the system's ability to retrieve all relevant documents, regardless of the number of irrelevant documents retrieved. In other words, it assesses the coverage or the completeness of obtained results. In our context, it is defined as the ratio of relevant documents retrieved to the total number of relevant documents, as shown in Eq. 4

$$\text{Recall} = \frac{\text{relevant verses retrieved}}{\text{total number of relevant verses}} \tag{4}$$

High recall indicates that the information retrieval system is effective in identifying most of the relevant documents. Conversely, high precision indicates that the system excels in excluding irrelevant documents.

F-score: the harmonic mean of recall and precision. This metric is particularly useful for balancing the trade-off between precision and recall. Mathematically, it's calculated using Eq. 5

$$F\text{-score} = \frac{2 \cdot \text{precision} \cdot \text{recall}}{\text{precision} + \text{recall}} \tag{5}$$

2) Gold Standard

Since no gold standard 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, as described in Subsection A of this section to create our gold standard. First, we selected only the most strongly related verse pairs from the QURV dataset (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 III.

TABLE III. Gold standard characteristics

Topic	Number of verses
Monotheism / ألتوحيدْ	35
Resurrection/ القيامة	150
Righteousness of parents/ بر الوالدين	8
Astronomy/ الفلك	21
Charity/ الصدقة	31
Isa ibn Maryam/ عيسى ابن مريم	25
Prayer/ لصلاة	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 IV presents the performance of the proposed approach in terms of precision, recall, and F-score.

As shown in Table IV, 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 system exhibited its lowest accuracy on the "Resurrection" topic, achieving an F-score of 43.42%. In this case, the system retrieved only 43 of 150 relevant verses, though it maintained a high precision of 89.58%.

Interestingly, the system's performance varied significantly across different topics. While the Flair model exhibited exceptional precision in capturing the semantic content of verses related to 'Isa ibn Maryam,' it encountered significant challenges in effectively addressing the 'Resurrection' topic. This discrepancy suggests that the model's effectiveness may be influenced by the nature of the topic, mainly how explicitly the topic is discussed within the verses.

The lower accuracy observed in the 'Resurrection' topic could be attributed to the brevity and complexity of its related verses. These verses typically encapsulate profound meanings in just a few words, which may not provide sufficient context for the model to capture their semantic information accurately.

TABLE IV. Performance of the proposed 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%**

Overall, the proposed system demonstrates an average precision of 90.79%, underscoring its effectiveness in retrieving semantically related verses while successfully excluding irrelevant ones. However, the variability in system performance across different topics suggests that further refinement is needed, especially for more complex topics like "resurrection".

To explore the impact of discourse segmentation process, we repeated the first set of experiments, but using the original Quran dataset [24] instead of Quran discourse units dataset. Table V shows the result of this experiment.

From Table V, 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 system's ability to retrieve all semantically related verses to the input query has considerably decreased when considering verses as atomic units. However, the resurrection topic experienced only a marginal degradation in precision and recall, estimated at 6.25% and 2%, respectively. This is because most related verses in this topic were not subjected to the discourse segmentation process, as they are typically very short.

These results underscore the significant potential of Quran discourse segmentation in enhancing the accuracy of semantic search. By segmenting verses based on recitation

TABLE V. Performance of the proposed approach without using discourse segmentation

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%
الصلاة/Prayer	57.89%	55%	56.41%
قصة عيسى ابن مريم The story of Jesus, son of Mary	73.68%	56%	63.63%
Average	**61.11%**	**49.5%**	**52.03%**

stop marks, our approach effectively captures the diverse topics embedded within the verses, which traditional models often overlook. This suggests that discourse segmentation could be a valuable technique for the Holy Quran and other religious texts with complex semantic structures. Additionally, the effectiveness of Flair model further highlights the importance of contextualized word embedding in capturing rich semantic content. This points to the potential for future research to explore even more advanced models, such as transformer-based models or large language models (LLMs), to achieve deeper semantic comprehension. These advancements could significantly influence the development of more intelligent and effective semantic search systems.

In summary, our findings successfully address the primary research objective of improving semantic search accuracy in the Holy Quran. Discourse segmentation combined with Flair model has proven effective, although further work is needed to enhance performance across all topics.

4) Comparison With Other Methods

To demonstrate the effectiveness of the proposed approach, it was crucial to conduct a comparative analysis with existing studies in the field. Accordingly, we selected the most recent semantic search tools for comparison: the semantic search approach based on Large Language Models (LLM) [16] and the Quranic Semantic Search Tool (QSST) [3], which are already discussed in the related work section. This comparison involved testing our approach using the same queries employed to evaluate these competing methods, given the absence of a standardized evaluation dataset for this task.

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'. These queries were also used to evaluate our semantic search approach. Evaluation results for each query in terms of precision are depicted in Figure 7, while Table VI provides a performance comparison in terms of average precision between the proposed approach and QSST.

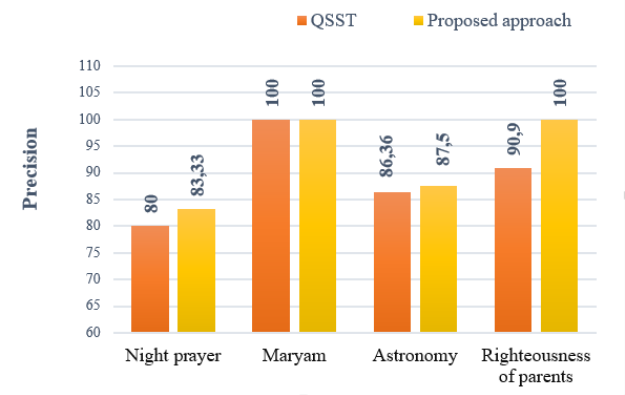

Figure 7. Performance Comparison in Terms of Precision of the Proposed Approach against QSST

TABLE VI. 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 VI, 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.

In the second experiment, the proposed approach was compared against the recently published semantic search approach based on LLM [16]. The author distinguishes three levels of semantic search: low-level, mid-level, and high-level, which correspond to searching by terms, by concepts, and by topics, respectively. For each level, a list of appropriate queries was used for evaluation. In our experiment, the same queries for each level were provided individually as input to our semantic search tool, and the resulting outputs were evaluated by five qualified Islamic experts who were previously involved in constructing our gold standard. The results, depicted in Figure 8, highlight the average precision of each approach.

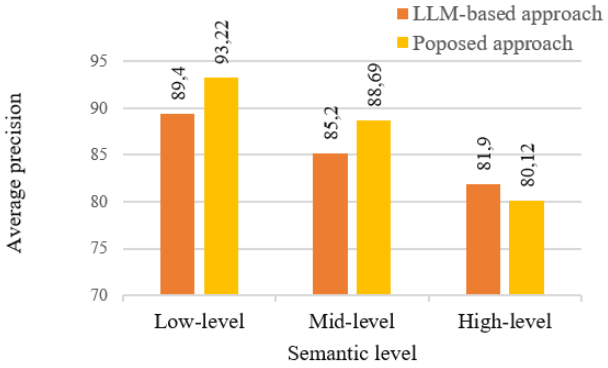

Figure 8. Performance Comparison in Terms of Precision of the Proposed Approach Against LLM-Based Approach

Figure 8 reveals that the precision of the proposed approach surpasses that of its competitor for both low-level and mid-level semantics, with average precisions estimated at 93.22% and 88.69%, respectively. However, when the semantic complexity of the queries increases, the LLM-based approach exhibits superior performance compared to our method. This suggests that the contextualized word representation Flair model requires more extensive classical Arabic data for training to effectively capture deep contextual dependencies among words in the Holy Quran. Overall, the results are very encouraging and demonstrate the effectiveness of the proposed approach.

5) Limitations

Despite the promising results, the proposed approach has some limitations that warrant consideration. First, its effectiveness is highly dependent on the accuracy of discourse segmentation; if the segmentation is not precise, the meanings of the verses can be misrepresented. Additionally, while Flair model enhances contextual understanding, it may still fall short of capturing the deeper and more complex semantic dependencies in the Holy Quran. Moreover, the scalability of this approach to other religious texts relies on the availability of well-defined domain-specific discourse markers, which may not always be present. Finally, the absence of a standardized evaluation dataset for semantic search in the Holy Quran poses challenges in benchmarking and comparing results with other studies.

In the next section, we summarize our contributions and suggest directions for future work.

5. Conclusions and Future Work

This paper proposes a new semantic search approach for the Holy Quran. The proposed approach begins by segmenting the Quranic text into non-overlapping discourse units based on specific recitation stop marks. These discourse units are then transformed into semantic vectors that capture their underlying meanings. To select the most suitable embedding model for this task, we investigated several embedding models for assessing semantic relatedness in the Holy Quran. We trained contextualized and non-contextualized embedding models on a large corpus of classical Arabic text. Subsequently, we evaluated and compared their performance in detecting semantically related verses in the Holy Quran. Evaluation results indicate that the Flair model achieves the highest performance. Consequently, this model was selected to convert the Quranic discourse units and the user input query into semantic vectors. Next, cosine similarity between the input query and all discourse unit vectors was computed to assign an appropriate score to each verse. Finally, top-ranked verses are chosen as the most semantically related verses to the input query. The evaluation results are auspicious, clearly demonstrating the effectiveness of the proposed approach.

A significant advantage of the proposed approach is its scalability and adaptability to other religious texts (such as Hadith books), provided that the appropriate domain-specific discourse markers are chosen. The principal findings of this research can be summarized as follows:

1) Segmenting Quranic discourse based on recitation stop marks is highly recommended for effectively addressing the challenge of semantic search in the Holy Quran.
2) Incorporating discourse analysis information can significantly improve the accuracy of semantic search, particularly for the Holy Quran.
3) Contextualized word representation models outperform traditional word embedding models in capturing word meanings in the Holly Quran. However, extensive datasets are required for training to capture deep contextual dependencies.
4) The rich semantics and profound meanings of the Holy Quran necessitate highly advanced AI models to effectively capture contextual dependencies and semantic nuances relating its words and verses.

In future work, we plan to enhance the performance of the proposed approach by utilizing advanced transformer-based models. Additionally, we will focus on developing an extensive evaluation dataset that can serve as a gold standard for semantic search in the Holy Quran. This initiative encourages researchers to explore this promising field further and benchmark their findings against a standardized evaluation framework.

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. 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. Baquee, M. Sharaf, and E. Atwell, "Qursim: A corpus for evaluation of relatedness in short texts," in *Proceedings of the Eighth International Conference on Language Resources and Evaluation (LREC'12)*. European Language Resources Association (ELRA), 2012, pp. 2295–2302.

[7] P. R. Christopher D. Manning and H. Schütze, *Introduction to information retrieval*. Cambridge University Press, 2008.

[8] A.-B. Sharaf and E. Atwell, "Qurana: Corpus of the quran annotated with pronominal anaphora," in *Proceedings of the 8th International Conference on Language Resources and Evaluation (LREC 2012)*. European Language Resources Association (ELRA), 2012, pp. 130–137.

[9] I. A. Mohammed Akour, Izzat Alsmadi, "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 embedding," in *2020 International Conference on Emerging Trends in Smart Technologies (ICETST)*. IEEE, 2020, pp. 1–6.

[11] M. Alqahtani and E. Atwell, "Arabic quranic search tool based on ontology," in *Natural Language Processing and Information Systems*. Springer, 2016, pp. 478–485.

[12] H. Afzal and T. Mukhtar, "Semantically enhanced concept search of the holy quran: Qur'anic english wordnet," *Arabian Journal for Science and Engineering*, vol. 44, no. 4, pp. 3953–3966, 2019.

[13] M. Haromainy, A. Sari, D. A. Prasetya, M. Subhan, A. Lisdiyanto, and T. Septianto, "Enhancing thematic holy quran verse retrieval through vector space model and query expansion for effective query answering," in *2023 IEEE 9th Information Technology International Seminar (ITIS)*. IEEE, 2023, pp. 1–6.

[14] R. A. Rajagede, K. Haryono, and R. Qardafil, "Semantic retrieval for indonesian quran autocompletion," *Jordanian Journal of Computers and Information Technology (JJCIT)*, vol. 9, no. 02, pp. 94–106, 2023.

[15] A. Tubaishat, S. Razzaq, F. Maqbool, M. Ilyas, and M. S. Khan, "Quran wordnet: A framework for semantic and sentiment search," in *2023 7th IEEE Congress on Information Science and Technology (CiSt)*, 2023, pp. 24–32.

[16] M. Alqarni, "Embedding search for quranic texts based on large language models," *The International Arab Journal of Information Technology (IAJIT)*, vol. 21, no. 02, pp. 243 – 256, 2024.

[17] S. Lagrini, N. Azizi, and R. Mohamed, "Exploiting discourse relations to produce arabic extracts," *International Journal of Reasoning-based Intelligent Systems*, vol. 14, no. 2-3, pp. 130–143, 2022.

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

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

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

[21] 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.

[22] 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.

[23] A. Akbik, D. Blythe, and R. Vollgraf, "Contextual string embeddings for sequence labeling," in *Proceedings of the 27th international conference on computational linguistics*. Association for Computational Linguistics, 2018, pp. 1638–1649.

[24] "Tanzil documents," available online, accessed 13.12.2023.

[25] M. Alrabiah, A. Alsalman, E. Atwell, and N. Alhelewh, "Ksucca: a key to exploring arabic historical linguistics," *International Journal of Computational Linguistics*, vol. 5, no. 2, p. 27, 2014.

[26] 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)*. Association for Computational Linguistics, 2021, pp. 176–185.

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

[28] 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*. Association for Computational Linguistics, 2016, pp. 11–16.

[29] 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.

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

[31] S. Arora, Y. Liang, and T. Ma, "A simple but tough-to-beat baseline for sentence embeddings," in *International Conference on Learning Representations*. ICLR, 2017.

[32] H. Al-Bataineh, W. Farhan, A. Mustafa, H. Seelawi, and H. Al-Natsheh, "Deep contextualized pairwise semantic similarity for arabic language questions," in *Proceedings of the 2019 IEEE 31st International Conference on Tools with Artificial Intelligence (ICTAI)*. IEEE, 2019, pp. 1586–1591.