# Sequran: Composite Scoring and Reranking Techniques for Refining Quran Search Engine Results

Ray Ramadita[1*], Wisnu Uriawan[2], and Wildan Budiawan Zulfikar[3]
[1−3]Informatics Department, Faculty of Science and Technology, UIN Sunan Gunung Djati Bandung
Jawa Barat, Indonesia 40614
Email: [1]rayramadita12@gmail.com, [2]wisnu_u@uinsgd.ac.id, [3]wildan.b@uinsgd.ac.id



***Abstract*—Information Retrieval (IR) systems are crucial in the development of accurate and relevant search engines, especially in domain-specific applications such as Sequran. While fine-tuning is a common optimization approach, this method often requires significant computational resources, diverse datasets, and time-consuming hyperparameter tuning, with the risk of performance degradation. To address these challenges, this research introduces and evaluates a novel architecture that enhances search relevance by integrating lexical and semantic signals, offering a practical alternative to resource-intensive fine-tuning. The proposed architecture involves a two-stage process. The first stage, composite scoring, enhances term frequency (BM25S) with a semantic intent booster. The second stage utilizes a Cross-Encoder (jina-reranker-v2-base-multilingual) to refine the ranking based on contextual relevance. Evaluation is conducted on a domain-specific Islamic query-answer dataset consisting of approximately 20 queries and around 18,000 text entries (over 6,000 verses, each paired with two tafsir types). The proposed method achieves Precision@10 = 0.230 and Recall@10 = 0.564 over the fine-tuned baseline (0.167/0.423), representing absolute gains of 6.3 and 14.1 percentage points, equivalent to relative improvements of 37.7% and 33.3%. Gains are consistent across both metrics, with the larger recall gain indicating that reranking surfaces relevant passages ranked below the cut-off by lexical matching alone. This improvement demonstrates a clear trade-off, as the total execution time increased to approximately 18.5 seconds, which may constrain real-time use. The main implication of this research is the validation of a practical architecture for improving IR systems, offering a viable alternative for domain-specific contexts such as Sequran.**

***Index Terms*—Composite Scoring, Reranking Techniques, Information Retrieval, Search Engine, Quran**

## I. Introduction

In recent years, extensive research on search engines has explored algorithmic and optimization approaches. However, many widely used systems still rely on traditional rule-based algorithms and keyword matching [1, 2] or TF-IDF models [3–5], primarily focusing on surface-level term overlap. These approaches remain effective for simple fact-based retrieval but often struggle with queries requiring deeper semantic understanding, nuanced intent interpretation, or contextual inference. As a result, they frequently return results that match keywords but fail to reflect the underlying meaning of the user's information need.

Examples such as Alfanous [6], Search Truth [7], OpenQuran, QuranSearch, Search the Quran, Lafzi [8–10], and Al-Hadees remain focused on lexical matching within the Quranic domain. Their approaches often struggle to interpret user intent, producing limited or inaccurate results. Accurate Quranic retrieval is critical not only for academic research but also for everyday learning and interpretation, where misleading results risk theological misunderstanding. These limitations have driven new researchs to enhance accuracy and contextual understanding in search systems.

Research in Information Retrieval (IR) has since evolved toward hybrid methods that combine multiple matching strategies to improve ranking accuracy [11, 12] This direction emerges from the realization that no single signal, lexical, semantic, or behavioral factor can consistently capture relevance across diverse query types. This principle underlies composite scoring, where diverse relevance indicators such as term frequency, retrieval confidence, document credibility, and user interaction patterns are integrated into a unified score [13–18]. Through this multi-signal integration, composite scoring enables systems to capture richer aspects of relevance and reduce the limitations of relying on any single matching mechanism.

The rise of deep learning shifted IR research toward model optimization. For instance, Zhan et al. [19] have proposed dynamic hard-negative sampling, enables the model to distinguish between documents that are very

Received: June 26, 2025; received in revised form: Nov. 12, 2025; accepted: Nov. 12, 2025; available online: Sep. 08, 2026.
*Corresponding Author

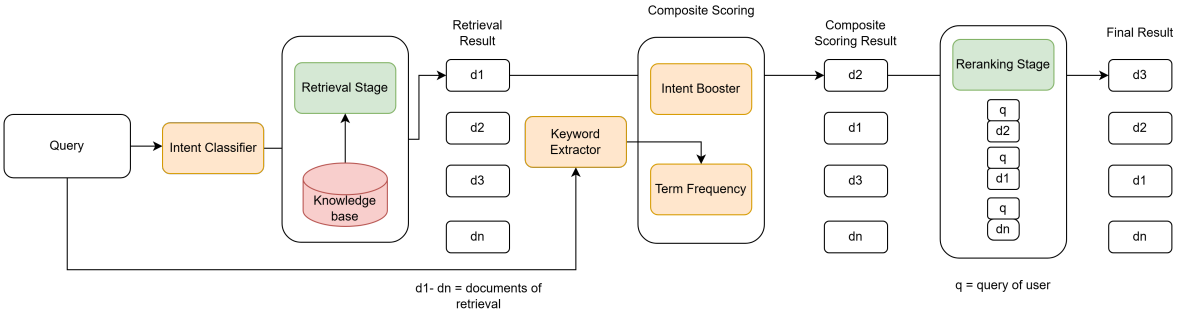


Fig. 1. Sequran’s workflow.

similar but irrelevant, improving dense retrieval quality. Deep architectures also introduce two-stage retrieval, where a reranking stage refines initial results. Cross-encoders, in particular, enhance ranking accuracy by analyzing query-document interactions at the token level, as shown in [20], for health IR.

Further progress comes through fine-tuning, adapting large pre-trained models with domain data to achieve state-of-the-art accuracy. Bidirectional Encoder Representations from Transformers (BERT)-based fine-tuning has improved IR and entity detection in archaeology [21] and biomedical tasks [22], achieving superior results through joint pre-training and fine-tuning. However, fine-tuning demands high computational cost, extensive data, and careful hyperparameter tuning [23, 24]. It also risks catastrophic forgetting, the loss of general knowledge after domain-specific training [25–28].

As a lighter alternative, hybrid pipelines combining lexical and semantic retrieval (e.g., BM25 with vector-based search [15, 29]) have been explored. However, most systems still rely on simple score aggregation without considering intent or context in the reranking process. This limitation is particularly evident in Quranic search, where both lexical accuracy and semantic depth are crucial.

In contrast, the proposed two-stage refinement architecture integrates lexical, semantic, and intent-aware signals within a composite scoring stage, followed by a reranking stage that models deeper contextual relevance. This layered refinement enables the system to combine the efficiency of hybrid pipelines with the interpretive precision typically associated with fine-tuned models. By first filtering candidates using lightweight scoring and applying a more expressive cross-encoder, the architecture addresses both scalability requirements and the need for theological or conceptual depth, an important factor for Quran retrieval tasks.

This research builds on previous Sequran research [30] which has performed strongly on public benchmarks, achieving Precision@10/Recall@10 scores of 0.4469/0.4477 on MIRACL [31] and 0.089/0.89 on TyDi [32, 33]. However, its performance has dropped substantially on domain-specific Islamic queries [34] (Precision@10 = 0.167, Recall@10 = 0.423). The low precision indicates that while recall remains moderate, many retrieved verses are false positives, suggesting that fine-tuning is highly sensitive to training data quality, domain mismatch, and input context length. This gap highlights the need for a retrieval pipeline that can maintain generalization while handling the conceptual and theological nuances of religious queries.

Therefore, this research systematically evaluates an enhanced Sequran pipeline to improve domain-specific retrieval relevance. This study provides three key contributions. First, it extends the original Sequran architecture with a composite lexical-semantic refinement pipeline capable of capturing both explicit and implicit relevance signals. Second, it empirically demonstrates consistent improvements in Precision@10 and Recall@10 over fine-tuned and lexical-only baselines, indicating that the refinement pipeline effectively addresses prior weaknesses in conceptual matching. Third, it highlights the efficiency trade-offs inherent in hybrid retrieval, offering insights into how relevance and latency can be balanced for domain-specific information retrieval tasks.

## II. Research Method

The overall workflow of the proposed retrieval pipeline is shown in Fig. 1. As seen in Fig. 1, q denotes the user query; d1, d2, . . . , dn denote the candidate documents returned from the knowledge base, where n is the number of retrieved candidates and the index indicates the rank at each stage. The pairs (q, di) in the reranking stage indicate that the query is compared

TABLE I
INTENT CATEGORIES IN THE RESEARCH.

| Intent | Description |
|---|---|
| Manners and Ethics | How to behave politely, respect others, and uphold Islamic moral values in daily life. |
| *Tawhid* and Divinity | Belief in Allah as the only God, including *tawhid uluhiyah*, *rububiyah*, and *asma wa sifat*. |
| Law and *Fiqh* | The rules of Islamic law that govern aspects of worship and muamalah are based on the Quran, *sunnah*, and scholarly *ijtihad*. |
| Social and Community Relations | Islamic principles and guidelines in maintaining social harmony and human relations. |
| Prophetic Stories, Advice, and Wisdom | Inspirational stories from the lives of the prophets that contain moral messages and wisdom. |
| Commandments and Restrictions | Islamic rules that guide people to do good and avoid bad. |
| Practices and Prayers | Daily worship and prayers taught in Islam to get closer to Allah. |
| *Akidah* and *Akhlak* | Beliefs and behaviors that reflect Islamic faith and morals. |
| Education and Science | The importance of studying and spreading knowledge in Islam. |
| Apocalypse and Warning | End-time events and warnings to prepare for the day of reckoning. |

Note: only the descriptions are vectorized and stored in a vector database for classification.

directly with each candidate document by the cross-encoder to produce the final ranking.

The process begins with intent classification, followed by an initial retrieval step that generates candidate verses from the knowledge base. These candidates are enriched using keyword extraction and intent boosting, and the final ranking is produced through a reranking stage. The following subsections describe each module in detail.

### A. Intent Classification

The first step in the proposed architecture is intent classification. To perform the task, this research leverages the fine-tuned Sentence-BERT Indonesia Islamic Sentence-BERT V2 model which allows the system to perform semantic classification without a need to retrain the model with a labeled dataset. Later, the model is used not only to classify the intent of user queries, but also to assign intents to the Quran and Tafsir datasets used as the knowledge base, supporting the intent booster process. Sentence-BERT has also been used for classification tasks [35].

According to previous studies [36–38], in the context of search engines, intent is generally divided into three types, namely navigational, informational, and transactional. However, to adapt to the context of Islam and the Quran, this research uses intent that is different from that mentioned before. The number of intent categories used in this research is limited to 10 types which can be seen in Table I. These categories are adapted from a book [39] and formulated in Indonesian for use in real-world applications.

After assigning intent labels to both the query and the knowledge base, each intent category is paired with a short textual description for semantic processing. Subsequently, these descriptions are embedded and stored in the vector database, while the intent labels function only as categorical tags. This procedure is summarized in Algorithm 1.

**Algorithm 1** Create Intent Embedding Collection

```
Require: intent_list: list of intents with their descriptions
Ensure: intent embedding collection in Qdrant
  function INTENT_EMBEDDING(intent_list)
      for all intent ∈ intent_list do
          intent_embedding ← EMBED(preprocessed intent description)
          INSERT_DOCUMENT(intent_embedding, intent)
      end for
      return
  end function
```

After the intent embedding process is complete, the system can perform intent classification on queries and Quran and Tafsir documents using the Sentence-BERT model (Indonesia Islamic Sentence-BERT V2). In this research, each query or document classified intent only produces intent that has a similarity value $\geq 0.5$. If the resulting similarity value is $< 0.5$, or the number of detected intent is $< 2$, the system automatically selects 3 intent with the highest score. The system process in classifying intent, both for queries and documents can be represented into Algorithm 2.

**Algorithm 2** Intent Classifier

```
Require: text: user query or document
Ensure: final_intent: list of intents
  function INTENT_CLASSIFIER(text)
      text_embedding ← EMBED(preprocessed text)
      temp_intent ← SEARCH_NEAREST_INTENT(text_embedding)
      final_intent ← empty list
      if intent_score of temp_intent ≥ threshold (0.5) then
          final_intent ← temp_intent
      end if
      if |final_intent| < 2 then
          final_intent ← top 3 intents from temp_intent
      end if
      return final_intent
  end function
```

### B. Keyword Extraction

Keyword extraction in information retrieval can be performed using two general approaches: linguistic-based methods and algorithmic methods. Linguistic approaches typically rely on Part-of-Speech (POS)

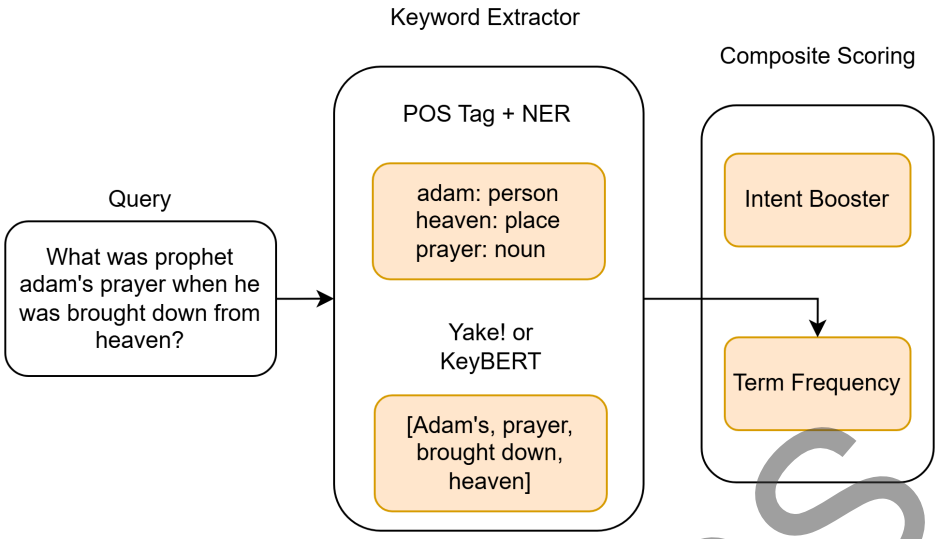


Fig. 2. Illustration of keyword extraction from a user query using two approaches: (1) Part-of-Speech (POS) tag and Named Entity Recognition (NER) based on query intent, and (2) statistical or contextual algorithms like YAKE! or KeyBERT. The extracted keywords are used in term frequency.

TABLE II
MAPPING OF TAGS AND ENTITIES PRIORITIZED FOR EACH INTENT.

| Intent | Tags | Entities |
|---|---|---|
| Manners and Ethics | ADJ, NOUN, VERB | PER, REG, GPE |
| *Tawhid* and Divinity | NOUN, PROPN, VERB | PER, REG |
| Law and *Fiqh* | NOUN, VERB, ADJ, NUM | LAW, QTY, MON, PRC, PER |
| Social and Community Relations | NOUN, VERB, PRON | PER, REG, ORG, GPE, MON |
| Prophetic Stories, Advice, and Wisdom | NOUN, VERB, PROPN | PER, EVT, GPE, LOC, DAT, TIM, REG, FAC |
| Commandments and Restrictions | VERB, NOUN, ADV | PER, REG |
| Practices and Prayers | NOUN, VERB, ADJ, PROPN, ADV | TIM, DAT, EVT, PER, LOC |
| *Akidah* and *Akhlak* | NOUN, ADJ, VERB | PER, REG |
| Education and Science | NOUN, ADJ, VERB | PER, ORG, GPE, FAC |
| Apocalypse and Warning | NOUN, VERB, ADJ | EVT, PER, REG |

Note: NOUN = common noun; PROPN = proper noun; VERB = verb; ADJ = adjective; ADV = adverb; PRON = pronoun; NUM = numeral; PER = person; ORG = organization; GPE = geopolitical entity (city, country); LOC = non-political location; FAC = facility/structure; REG = religion; LAW = legal ruling or statute; EVT = event; DAT = date; TIM = time; MON = money; PRC = percent/fraction; QTY = quantity with unit. Part-of-Speech (POS) tags follow the Universal POS (UPOS) tag set of Universal Dependencies, produced by a Flair sequence tagger trained on the UD Indonesian-GSD treebank, and Named Entity Recognition (NER) tags follow the 19-type NERGRIT label set used by cahya/bert-base-indonesian-NER.

tagging and Named Entity Recognition (NER) to identify meaningful terms [40–42] while algorithmic approaches extract keywords directly from raw text. Examples of the latter include YAKE! [43] which has been used in previous IR research [44, 45], and KeyBert [46], which has been widely adopted in various NLP applications [47–50]. The overall process adopted in this research is illustrated in Fig. 2.

This research uses the BERT-base Indonesian NER model to predict entities [51] and the Flair NLP model (with slight modifications and adjustments using Indonesian language) to predict POS tags [52, 53]. Then, to ensure the relevance of the extracted keywords, each user intent is mapped to a specific set of tags and entity types. During the term frequency calculation, only the extracted keywords whose tags or entities match this pre-defined set for the given intent are considered for scoring. This targeted extraction mechanism is detailed in Table II, which presents the complete mapping for all intents used in this research.

The second approach, like YAKE! is quite simple as it can be configured specifically. This research uses the following configurations: Indonesian (lan =‘id’) as the base and stopwords removal; n-grams maximum 3 (n= 3); and top = 5. All other parameters are left at the default values provided by YAKE!

Meanwhile, KeyBERT can directly use the same model as used in vector database when encoding

data [54]. However, the mechanism is similar to POS tag and NER that call a new model. The concept used by KeyBERT is to create a new process to run the model even though the model used is the same. All approaches are illustrated in Fig. 2.

### C. Vector Database

In another Sequran research [30], all vector representations of Quranic verses and tafsir are stored as raw tensor files. This approach complicates future scalability, slows down indexing, and limits efficient retrieval operations. To address these constraints, this research employs a Vector Database Management System (VDBMS) to store and manage vector embeddings, enabling more scalable development and faster retrieval in subsequent system iterations.

Several VDBMS options exist, including Milvus, Qdrant, Weaviate, and Chroma [55]. This research adopts Qdrant due to its strong support for complex query processing and its efficient pre-filtering capabilities using attribute indexing or Hierarchical Navigable Small World (HNSW)-based predicate filtering. These features, combined with Qdrant's adaptive query-planning based on predicate selectivity [56], substantially reduce the search space and improve retrieval performance.

After selecting Qdrant as the VDBMS, the translation and tafsir-tahlili entries [34] are encoded into vector embeddings using the Indonesia Islamic Sentence-BERT V2 model. Before uploading these embeddings into Qdrant, each entry is assigned an intent label using Algorithm 2 to support the intent booster mechanism used later in the pipeline. Once annotated, the vectors and their metadata are inserted into a Qdrant collection, configured with parameters listed in the Evaluation section. The collection uses cosine similarity as the distance metric, aligned with the embedding characteristics of the selected model.

### D. Composite Scoring

The composite scoring architecture used in this research integrates two relevance signals: term frequency and intent booster. BM25 is employed as the term-frequency component because its length-normalization mechanism is effective for handling the varying document lengths found in Quran translations and tafsir. It allows BM25 to provide more stable lexical relevance scores before the refinement and reranking stages [29].

There are two ways to implement BM25. The first is to use a modified Python implementation, called BM25S [57], which can be represented in algorithmic form, as shown in Algorithm 3. Second, using hybrid-search by combining vector embedding from Sentence-BERT and sparse embedding BM25 from Qdrant into a vector database. The implementation of hybrid-search is similar to the explanation in vector database or refer to the Qdrant hybrid-search documentation with minor modifications and adjustments.

**Algorithm 3** Calculate Term Frequency Using BM25S

**Require:** query: user query; hits: list of hits from initial retrieval
**Ensure:** new_hits: updated hits with BM25S scores
**function** TERM_FREQUENCY(query, hits)
  corpus ← preprocessed `translate` + `tafsir_tahlili`
  corpus_tokens ← BM25TOKENIZER(corpus)
  retriever ← BUILD_BM25S_RETRIEVER(corpus)
  retriever_index ← BUILD_INDEX(corpus_tokens)
  query_tokens ← BM25TOKENIZER(query)
  _, scores ← RETRIEVE(retriever_index, query_tokens, $k = |\text{hits}|$)
  new_hits ← empty list
  $i \leftarrow 0$
  **for all** hit ∈ hits **do**
    hit.score ← hit.score + scores$[0, i]$
    new_hits ← new_hits + [hit]
    $i \leftarrow i + 1$
  **end for**
  **return** new_hits
**end function**

The main difference between the two implementations lies in the placement of the BM25 computation within the retrieval pipeline. In the BM25S approach, lexical scoring is applied only after the top-k documents are retrieved, functioning as a refinement layer. In contrast, Qdrant's hybrid-search performs BM25 and vector-search independently and fuses their results using Reciprocal Rank Fusion (RRF), which has been shown to improve retrieval robustness by combining complementary relevance signals [58].

To calculate the overall composite scoring, this research develops a formula adapted from Google's patent [59] as an intent booster component. This component will later be summed up with the initial score to produce a more optimal final score. The formula used can be written as Eq. (1):

$$\text{booster} = W + \alpha \times \hat{S}_h + \beta \times S_i + \gamma \times \hat{S}_h \times S_i. \tag{1}$$

It defines the composite score, which combines lexical, semantic, and intent matching factors. The $W$ is the weight of the number of intent matches between the query and the document (higher matches yield larger $W$ values). Then, $\hat{S}_h$ is the normalized hit score obtained by applying Eq. (2) to the raw hit score $S_h$, i.e., the semantic retrieval score (primary signal) further refined by BM25S as a secondary lexical booster, and $S_i$ is intent score derived from query-document intent similarity. Next, $\alpha, \beta$ are control weights. In this design, $\alpha$ is set larger than $\beta$, since the semantic score is the main retrieval signal, while the intent score is used only as a complementary booster. Meanwhile, $\gamma$

is dynamic factor obtained by scaling down the raw hit score $S_h : \gamma = \frac{S_h}{10}$ if $S_h > 1$, and $\gamma = \frac{S_h}{5}$ otherwise.

To ensure the final score is consistent and balanced, the hit score in Eq. (1) is normalized to a range between 0 and 1. It prevents any single criterion from dominating the composite score and allows for proportional weighting [13]. The formula for normalization (Eq. (2)) can be denoted in the following form:

$$\hat{S}_h = \frac{S_h - Min\_score}{Max\_score - Min\_score}, \tag{2}$$

where $S_h$ is the raw hit score, and $Min\_score$ and $Max\_score$ are the smallest and largest scores among the retrieved hits. Then, the Eq. (1) and (2) can be represented into Algorithm 4.

**Algorithm 4** Intent Booster

```
Require: intents: list of intents; hits: list of hits from initial retrieval
Ensure: boosted_hits: updated hits with boosted scores
  function INTENT_BOOSTER(intents, hits)
    boosted_hits ← empty list
    α ← 0.5;  β ← 0.3
    all_scores ← [h.score for h in hits]
    min_score ← min(all_scores);  max_score ← max(all_scores)
    for all hit ∈ hits do
      normalized_score ← NORMALIZE(hit.score, min_score, max_score)
                                                        ▷ Eq. (2)
      document_intent ← intents associated with hit
      matching_intent ← document_intent ∩ intents
      match_count ← |matching_intent|
      intent_score ← sum of scores in matching_intent
      if hit.score > 1 then
        dynamic_score ← hit.score / 10
      else
        dynamic_score ← hit.score / 5
      end if
      if match_count ≥ 3 then
        W ← 0.6
      else if match_count = 2 then
        W ← 0.4
      else
        W ← 0.2
      end if
      booster ← W + α· normalized_score +β· intent_score ▷ Eq. (1)
      booster ← booster + dynamic_score · normalized_score · in-
      tent_score
      hit.score ← hit.score + booster
      boosted_hits ← boosted_hits + [hit]
    end for
    return boosted_hits
  end function
```

### E. Reranking

The final stage of this refinement architecture is a reranking process applied only to top-30 results from composite scoring. Instead of replacing the initial score as in general two-stage retrieval architectures, the reranking score here is added to the initial composite score as a final contextual refinement. This mechanism works by evaluating each query-document pair individually to produce a new, more accurate relevance score that combines the initial relevance signal with a deep contextual understanding.

To maintain the efficiency of the reranking process, this research used the tafsir-wajiz (summary text) column as document input. It is because of the reranker model architecture, where the computational cost will increase along with the length of the input. By using tafsir-wajiz, the system can make sure the reranking process stays efficient without losing important semantic information.

The reranking models used in this research include several cross-encoder variants from MS-MARCO [60], as well as jina-reranker-v2-base-multilingual and bge-reranker-v2-m3. These models are selected based on their ability to understand Indonesian and multilingual languages to be compatible when used on the Quran and Tafsir dataset. All models are implemented using CrossEncoder class from Sentence-Transformers library.

### F. Evaluation

The evaluation in this research is conducted to see the effectiveness of various techniques and their combinations in improving the quality of search results and capturing more relevant results in Sequran with precision and recall metrics as a reference. In this research, execution time refers to the total wall-clock time required to process the entire evaluation dataset, measured from query input to ranked output generation. This metric is also analyzed to assess the practical efficiency of each technique and its combination, ensuring that the proposed approach remains feasible for production environments.

Each evaluation is tested at the @10 level of search results to ensure a better user experience because many search engine users only see around top-30 search results [61, 62]. It is also supported by Urman and Makhortykh [63], which have found that 97.11% of Google users and 99.49% of Bing users only view search results on the first page, with a majority (over 86% for Google and 92% for Bing) only clicking on the top-5 results. However, this research also includes evaluation results at the @30 and @100 search result levels for the needs of other researchers and future research.

Although standard IR evaluation datasets such as TREC [64], TyDi [32, 33], and MIRACL [31] are commonly used, the architecture proposed in this research has prerequisites that make it incompatible. Specifically, it requires a knowledge base in which each document is pre-annotated with intent categories (as in Table I), a feature that is not available in standard corpus data. Manually annotating the millions of documents in the dataset to fulfill this prerequisite is an impractical task. Therefore, to enable valid evaluation,

TABLE III
EXPERIMENTAL SETTINGS AND HYPERPARAMETERS FOR SEQURAN, INCLUDING TESTED KEYWORD EXTRACTION VARIANTS.

| Component | Parameter Setting | Value Description |
|---|---|---|
| Semantic Retrieval (Qdrant) | Top-k retrieved documents | 100 |
| | Similarity function | Cosine similarity |
| | Embedding model | Indonesia Islamic Sentence-BERT V2 |
| | Threshold | 0.3 |
| | Indexed columns | translation & tafsir-tahlili |
| Intent Classification | Number of intents | 10 |
| | Similarity threshold | Select intents $\geq 0.5$; if $< 2$ is found, pick top 3 by score |
| Keyword Extraction (Tested Variants) | Methods explored | Flair NLP (POS tag), BERT-base Indonesian NER, YAKE!, KeyBERT |
| | Description | Tested as alternative keyword extraction methods using POS tagging, statistical, and contextual embeddings. |
| | Usage in final pipeline | Excluded (performed worse than non-extraction setup) |
| Composite Scoring | $\alpha$ (alpha) | 0.5 |
| | $\beta$ (beta) | 0.3 |
| | $\gamma$ (dynamic factor) | Divide hit_score by 10 if $> 1$, otherwise by 5 |
| Reranking | Used column | tafsir-wajiz |
| | Model names | MS-MARCO variants, jina-reranker-v2-base-multilingual (used), bge-reranker-v2-m3 |
| | Top-k | 30 (for application inference), 100 (for testing) |

Note: Bidirectional Encoder Representations from Transformers (BERT), Part-of-Speech (POS), and Named Entity Recognition (NER).

this research utilizes the query set with its relevant documents from the dataset [34], as the identifiers of each document match the annotated knowledge base prepared for this research.

The dataset includes approximately 20 queries and around 18,000 text entries (consisting of 6,000+ translated verses, each paired with two Tafsir types: *tahlili* and *wajiz*), ensuring compatibility with the intent-based retrieval framework. All experiments utilizing this dataset are executed on a Windows 11 machine equipped with an Intel Core i7-11800H CPU, 16GB RAM, and an NVIDIA RTX 3060 Graphics Processing Unit (GPU) (6GB VRAM). All evaluation metrics are computed using the ranx library [65] with default settings. For reproducibility, the key experimental settings and hyperparameters used in this research are summarized in Table III.

## III. RESULTS AND DISCUSSION

### A. Composite Scoring Evaluation

The evaluation at this stage focuses on term frequency with intent booster as the baseline. After that, three keyword extraction approaches (YAKE!, KeyBERT, and POS tag + NER) are gradually added and compared. This analysis also compares two term frequency implementations, BM25 and BM25S, to determine which approach is superior.

The evaluation results in Table IV show that YAKE! is the most efficient approach in terms of execution time when using either BM25S or BM25. It is because it does not require a corpus or deep learning model like the POS or NER approach [54]. However, it is a drawback of YAKE! itself as it is not able to semantically analyze the text, which is a very important feature in the field of IR [66].

TABLE IV
COMPOSITE SCORING EVALUATION USING BM25S, BM25, AND INTENT BOOSTER (WITH/WITHOUT KEYWORD EXTRACTION), MEASURED AT TOP-10 RESULTS (PRECISION@10, RECALL@10, AND EXECUTION TIME).

| Model | Extraction | P@10 | R@10 | Execution Time (s) |
|---|---|---|---|---|
| BM2SS | YAKE! | 0.211 | 0.541 | **6.83** |
| | KeyBERT | 0.211 | 0.541 | 9.78 |
| | POS + NER | 0.211 | 0.541 | 9.53 |
| | None | **0.215** | **0.547** | 7.55 |
| BM25 | YAKE! | 0.152 | 0.416 | **2.11** |
| | KeyBERT | 0.148 | 0.406 | 4.00 |
| | POS + NER | 0.156 | 0.433 | 3.38 |
| | None | **0.163** | **0.480** | 2.19 |

Note: P@10 is Precision@10, R@10 is Recall@10, and bolded numbers highlight superior results. Part-of-Speech (POS) and Named Entity Recognition (NER).

Table IV also shows that BM25, whether using extraction techniques or not, produces superior execution times compared to BM25S, even though its precision and recall results are still far below. It occurs because BM25 (Qdrant) performs a parallel hybrid search, combining vector embedding and sparse embedding, and calculate the results with the RRF algorithm. Although the RRF approach is claimed to improve search quality [58], BM25S is proven to be superior in this research. It only acts as a post-processor that adds lexical scores while maintaining semantic quality. Hence, it avoids the risk of reduced precision and recall that occurs in BM25 (Qdrant) due to the combination of lexical signals at the retrieval stage. It also proves that the BM25 approach in general with the transformer model can complement each other, which has an impact on improving the relevance of search results [67] in [14].

Other findings indicate that eliminating keyword

TABLE V
EXTENDED COMPOSITE SCORING EVALUATION AT DEEPER CUTOFFS (PRECISION AND RECALL @30 AND @100.

| Model | Extraction | Precision | | Recall | |
|---|---|---|---|---|---|
| | | @30 | @100 | @30 | @100 |
| BM25S | YAKE! | 0.100 | 0.039 | 0.667 | **0.817** |
| | KeyBERT | **0.101** | 0.039 | 0.676 | **0.817** |
| | POS + NER | 0.100 | 0.039 | 0.667 | **0.817** |
| | None | **0.101** | 0.039 | **0.679** | 0.793 |
| BM25 | YAKE! | 0.086 | 0.034 | 0.612 | 0.722 |
| | KeyBERT | 0.084 | 0.033 | 0.620 | 0.745 |
| | POS + NER | 0.079 | 0.032 | 0.562 | 0.687 |
| | None | **0.099** | **0.036** | **0.673** | **0.759** |

Note: Bolded numbers highlight superior results. Part-of-Speech (POS) and Named Entity Recognition (NER).

extraction results in higher relevance scores, whereas previous research [68, 69] have found that keyword extraction generally improves IR performance, providing another perspective on related research. This difference can be attributed to the characteristics of the translated Quran and Tafsir texts, where religious concepts are often expressed through context-dependent phrases or terminology. Many of these expressions lose their semantic meaning when reduced to separate keywords, causing important relevance indicators to be filtered out during extraction. Therefore, in this domain, composite scoring based only on BM25S and intent booster already captures the dominant lexical and contextual signals, while additional extraction steps may introduce noise. Additional comparisons of composite scoring performance, measured by precision and recall at @30 and @100, are summarized in Table V.

### B. Reranking Evaluation

After establishing that BM25S without keyword extraction and intent booster provides the strongest baseline for composite scoring, the next step is to evaluate whether reranking can further improve retrieval quality. This step is important because reranking models often capture contextual nuances that may be missed by lexical scoring alone. Therefore, the combined pipeline is evaluated to determine whether it can produce higher precision and recall. The results are presented in Table VI.

As shown in Table VI, jina-reranker-v2-base-multilingual not only excels in precision and recall, but also in efficiency. It is worth noting that this performance improvement is achieved without using special optimizations such as flash-attention. This result demonstrates that the basic architecture of this model is already robust, enabling it to deliver low latency even when run on more common devices without requiring Compute Unified Device Architecture (CUDA) configuration or the latest GPU versions. This

TABLE VI
RERANKING PERFORMANCE OF VARIOUS CROSS-ENCODER MODELS, EVALUATED AT TOP-10 RESULTS.

| Model | P@10 | R@10 | Execution Time (s) |
|---|---|---|---|
| ms-marco-MiniLM-L-6-v2 | 0.159 | 0.359 | 16.3 |
| ms-marco-MiniLM-L-12-v2 | 0.167 | 0.367 | 27.3 |
| ms-marco-electra-base | 0.226 | 0.562 | 62.0 |
| jina-reranker-v2-base-multilingual | **0.230** | **0.564** | 18.5 |
| bge-reranker-v2-m3 | 0.226 | **0.564** | **451.0** |

Note: P@10 is Precision@10, R@10 is Recall@10, and bolded numbers highlight notable results for discussion.

TABLE VII
EXTENDED RERANKING EVALUATION AT DEEPER CUTOFFS (PRECISION AND RECALL @30 AND @100).

| Model | Precision | | Recall | |
|---|---|---|---|---|
| | @30 | @100 | @30 | @100 |
| ms-marco-MiniLM-L-6-v2 | 0.085 | 0.039 | 0.499 | 0.793 |
| ms-marco-MiniLM-L-12-v2 | 0.086 | 0.039 | 0.510 | 0.793 |
| ms-marco-electra-base | **0.107** | 0.039 | **0.702** | 0.793 |
| jina-reranker-v2-base-multilingual | 0.104 | 0.039 | 0.698 | 0.793 |
| bge-reranker-v2-m3 | **0.106** | 0.039 | **0.704** | 0.793 |

characteristic makes it a highly attractive and easily accessible option for other researchers.

Interestingly, the number of parameters in jina-reranker-v2-base-multilingual is only around 278M, which is much less than bge-reranker-v2-m3, with around 568M parameters. The jina-reranker-v2-base-multilingual with smaller parameters is not only superior in terms of execution time, but also produces the highest precision with equivalent recall with bge-reranker-v2-m3. This finding is also consistent with research [70] that a large number of parameters does not always guarantee better results. It demonstrates that the quality of a model's architecture can have a more significant impact than simply relying on scaling. Additional comparisons of reranking performance, measured by precision and recall at @30 and @100, are summarized in Table VII.

### C. Discussion

The previous evaluations demonstrate that the proposed architecture consistently improves contextual relevance across diverse query types. To further illustrate its practical effectiveness, a qualitative case study is conducted using the query "What is the origin of the universe according to the Quran?" (*Darimana asal mula alam semesta menurut Quran?*). This query is selected because it reflects a complex theological concept that cannot be fully captured through lexical matching alone. Table VIII compares the top retrieved verses produced by the fine-tuned baseline and the

TABLE VIII
COMPARISON OF RESULTS BETWEEN BASELINE AND PROPOSED SYSTEMS FOR QUERY TRIALS.

| Rank | Fine-tuned Baseline | Proposed |
|---|---|---|
| 1 | Qaf: 15 | Gafir: 57 |
| ... | ... | ... |
| 4 | Gafir: 57 | **Az-Zariyat: 47** |
| ... | ... | ... |
| 10 | Al-Ankabut: 20 | **Al-Anbiya: 30** |

Note: Bolded text indicates the most relevant verses based on *tafsir*.

proposed system, highlighting the differences in their interpretive depth and semantic alignment.

As shown in Table VIII, the fine-tuned baseline system primarily retrieves verses containing explicit lexical cues related to the sky or creation such as Qaf:15 or Gafir:57. In contrast, the proposed architecture elevates verses that are central within classical and contemporary *tafsir* discussions, most notably Al-Anbiya:30 (the separation of heaven and earth) and Az-Zariyat:47 (the expanding universe). These verses appear within the top 10 instead of the top 30, demonstrating that the multi-stage pipeline successfully identifies deeper conceptual relationships beyond surface-level textual overlap. This improvement is driven by the combination of lexical scoring, which amplifies key conceptual terms, and the reranker, which captures semantic and theological nuances that the baseline embedding model fails to detect.

Compared with earlier Quran retrieval researches, which primarily rely on keyword-based search or single-stage semantic embeddings, the proposed architecture offers several advantages. Keyword-based systems effectively retrieve verses containing explicit lexical matches but often conceptual or interpretive relationships that are more extensive. Instead, semantic-only approaches frequently return verses that are generally related but do not have a proper relationship with tafsir interpretations. The multi-stage pipeline developed in this research addresses these limitations by integrating lexical scoring, intent booster, and reranking. This design not only aligns with findings from previous research [71], highlighting the benefits of layered retrieval strategies, but also extends them to Quran search tasks that require a higher degree of conceptual reasoning and theological precision. Through this layered approach, the system captures both explicit lexical signals and implicit semantic meaning, making it more suitable for domain-specific queries that demand interpretive depth.

Beyond accuracy improvements, these findings highlight more extensive implication for religious information retrieval. Queries involving theological, conceptual, or cosmological themes often require models to understand relationships that transcend lexical similarity. The ability of the proposed architecture to prioritize verses central to discussion suggests that multi-stage retrieval pipelines are especially effective for religious texts, where meaning is shaped not only by words but also by rich interpretations.

Although this research emphasizes relevance improvements, enhanced search performance inevitably brings trade-offs. Multi-stage scoring and reranking increase computational cost relative to the fine-tuned baseline. While these additional steps lead to more meaningful retrieval results, they also influence runtime and scalability. For use cases such as academic research tools, digital libraries, or thematic Quran study platforms, deeper relevance may be more valuable than minimal latency. However, for interactive or real-time applications, efficiency constraints may necessitate lighter configurations. Understanding the balance between relevance, latency, and computational resources is therefore crucial for determining appropriate deployment strategies. Overall, the results prove that multi-stage retrieval pipelines can substantially improve semantic and theological relevance in Quran search systems. The architecture introduced in this research provides a promising foundation for future work aimed at optimizing efficiency while preserving interpretive depth, particularly for domain-specific queries that require nuanced understanding beyond lexical matching.

## IV. CONCLUSION

This research has demonstrated a multi-stage retrieval architecture that notably improves the performance of the Sequran application without relying on fine-tuning. The combination of BM25S lexical scoring, intent booster, and reranking using jina-reranker-v2-base-multilingual yields the most balanced trade-off between relevance and computational cost. Quantitatively, this configuration achieves a moderate yet meaningful improvement in retrieval performance, with Precision@10 increasing to 0.230 (+6.3 percentage points, or 37.7% in relative terms) and Recall@10 reaching 0.564 (+14.1 percentage points, or 33.3% in relative terms). These gains are accompanied by a higher computational cost, as the total execution time per evaluation increased to approximately 18.5 seconds. Despite this efficiency trade-off, the results represent a substantial advancement in developing a more context-sensitive and theologically aligned Quran IR system.

Several limitations should be acknowledged. The evaluation dataset is relatively small and constructed

through expert labeling, which introduces subjectivity and limits generalizability. Additionally, the multi-stage retrieval pipeline, particularly the reranker, adds latency that may challenge real-time use cases. These constraints indicate the need for broader validation and optimization.

Nonetheless, the findings offer practical value. The proposed architecture can be integrated into existing BM25-based or hybrid retrieval systems as a lightweight refinement layer, providing deeper conceptual relevance without the need for costly fine-tuning. It makes the approach suitable for religious, academic, and low-resource search applications where interpretive accuracy is crucial.

Future research may expand the evaluation dataset, refine latency through model or pipeline optimization, and explore semantic query expansion techniques to handle more complex conceptual queries. Further benchmarking against fine-tuned models will also clarify the trade-offs between efficiency and interpretive depth. It may open opportunities for combining fine-tuned models with multi-stage refinement pipelines.

## ACKNOWLEDGEMENT

This research was conducted as part of the first author's undergraduate thesis. Publication support is provided by the Informatics Department, Faculty of Science and Technology, UIN Sunan Gunung Djati Bandung. The authors gratefully acknowledged this support. No specific grant number was associated with this assistance.

## AUTHOR CONTRIBUTION

Conceived and designed the analysis, R. R.; Collected the data, R. R.; Contributed data or analysis tools, R. R.; Performed the analysis, R. R.; Wrote the paper, R. R.; and Provided critical feedback on language, sentence structure, references, and overall content, including suggestions for additions and removals to enhance the clarity and quality of the manuscript, W. U. and W. B. Z.

## DATA AVAILABILITY

The data that support the findings of the research are openly available in HuggingFace at https://huggingface.co/datasets/ramadita/Indo-Islamic-QA.

## REFERENCES

[1] B. K. Hussan, "Comparative study of semantic and keyword based search engines," *Advances in Science, Technology and Engineering Systems Journal*, vol. 5, no. 1, pp. 106–111, 2020.

[2] K. K. Dukhnovska and I. L. Myshko, "Analysis and comparison of full-text search algorithms," *Control Systems & Computers*, no. 3, pp. 45–52, 2024.

[3] M. M. Al Haromainy, A. P. 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)*. Batu Malang, Indonesia: IEEE, Oct. 18–20, 2023, pp. 1–6.

[4] S. E. Pratama, W. Darmalaksana, D. S. Maylawati, H. Sugilar, T. Mantoro, and M. A. Ramdhani, "Weighted inverse document frequency and vector space model for hadith search engine," *Indonesian Journal of Electrical Engineering and Computer Science*, vol. 18, no. 2, pp. 1004–1014, 2020.

[5] J. Chamorro-Padial, F.-J. Rodrigo-Ginés, and R. Rodríguez-Sánchez, "Finding answers to COVID-19-specific questions: An information retrieval system based on latent keywords and adapted TF-IDF," *Journal of Information Science*, vol. 50, no. 4, pp. 935–951, 2024.

[6] F. Beirade, H. Azzoune, and D. E. Zegour, "Semantic query for Quranic ontology," *Journal of King Saud University Computer and Information Sciences*, vol. 33, no. 6, pp. 753–760, 2021.

[7] S. Hakak, G. A. Gilkar, and W. Z. Khan, "Performance comparison of Qur'anic search engines," in *2020 International Conference on Computing and Information Technology (ICCIT-1441)*. Tabuk, Saudi Arabia: IEEE, Sept. 9–10, 2020, pp. 1–4.

[8] I. K. Fitriani, M. A. Bijaksana, and K. M. Lhaksmana, "Qur'an search system for handling cross verse based on phonetic similarity," *Jurnal Sisfokom (Sistem Informasi dan Komputer)*, vol. 10, no. 1, pp. 46–51, 2021.

[9] N. I. Purwita, M. A. Bijaksana, K. M. Lhaksmana, and M. Z. Naf'an, "Typo handling in searching of Quran verse based on phonetic similarities," *Register: Jurnal Ilmiah Teknologi Sistem Informasi*, vol. 6, no. 2, pp. 130–140, 2020.

[10] A. Octavia, M. A. Bijaksana, and K. M. Lhaksmana, "Verse search system for sound differences in the Qur'an based on the text of phonetic similarities," *Jurnal Sisfokom (Sistem Informasi dan Komputer)*, vol. 9, no. 3, pp. 317–322, 2020.

[11] J. Wang, M. Pan, T. He, X. Huang, X. Wang, and X. Tu, "A pseudo-relevance feedback framework combining relevance matching and semantic

matching for information retrieval," *Information Processing & Management*, vol. 57, no. 6, 2020.

[12] M. Wang, J. Liu, J. Wang, Y. Wang, and X. Chu, "A topicality relevance-aware intent model for web search," *IEEE Access*, vol. 11, pp. 65 739–65 748, 2023.

[13] M. C. Avornicului, V. P. Bresfelean, S. C. Popa, N. Forman, and C. A. Comes, "Designing a prototype platform for real-time event extraction: A scalable natural language processing and data mining approach," *Electronics*, vol. 13, no. 24, pp. 1–27, 2024.

[14] K. A. Hambarde and H. Proenca, "Information retrieval: Recent advances and beyond," *IEEE Access*, vol. 11, pp. 76 581–76 604, 2023.

[15] S. Bruch, S. Gai, and A. Ingber, "An analysis of fusion functions for hybrid retrieval," *ACM Transactions on Information Systems*, vol. 42, no. 1, pp. 1–35, 2023.

[16] A. Esteva *et al.*, "COVID-19 information retrieval with deep-learning based semantic search, question answering, and abstractive summarization," *npj Digital Medicine*, vol. 4, pp. 1–9, 2021.

[17] A. Kouadria, O. Nouali, and M. Y. H. Al-Shamri, "A multi-criteria collaborative filtering recommender system using learning-to-rank and rank aggregation," *Arabian Journal for Science and Engineering*, vol. 45, no. 4, pp. 2835–2845, 2020.

[18] M. Esposito, E. Damiano, A. Minutolo, G. De Pietro, and H. Fujita, "Hybrid query expansion using lexical resources and word embeddings for sentence retrieval in question answering," *Information Sciences*, vol. 514, pp. 88–105, 2020.

[19] J. Zhan, J. Mao, Y. Liu, J. Guo, M. Zhang, and S. Ma, "Optimizing dense retrieval model training with hard negatives," in *Proceedings of the 44th International ACM SIGIR Conference on Research and Development in Information Retrieval*. Association for Computing Machinery, July 11–15, 2021, pp. 1503–1512.

[20] R. Upadhyay, A. Askari, G. Pasi, and M. Viviani, "Beyond topicality: Including multidimensional relevance in cross-encoder re-ranking: The health misinformation case study," in *European Conference on Information Retrieval*. Glasgow, UK: Springer, March 24–28, 2024, pp. 262–277.

[21] A. Brandsen, S. Verberne, K. Lambers, and M. Wansleeben, "Can BERT dig it? Named entity recognition for information retrieval in the archaeology domain," *Journal on Computing and Cultural Heritage (JOCCH)*, vol. 15, no. 3, pp. 1–18, 2022.

[22] P. Su and K. Vijay-Shanker, "Investigation of improving the pre-training and fine-tuning of BERT model for biomedical relation extraction," *BMC Bioinformatics*, vol. 23, pp. 1–20, 2022.

[23] H. Soudani, E. Kanoulas, and F. Hasibi, "Fine tuning vs. retrieval augmented generation for less popular knowledge," in *SIGIR-AP 2024: Proceedings of the 2024 Annual International ACM SIGIR Conference on Research and Development in Information Retrieval in the Asia Pacific Region*. Tokyo, Japan: Association for Computing Machinery, Dec. 9–12, 2024, pp. 12–22.

[24] X. Ma, J. Guo, R. Zhang, Y. Fan, and X. Cheng, "Scattered or connected? An optimized parameter-efficient tuning approach for information retrieval," in *Proceedings of the 31st ACM International Conference on Information & Knowledge Management*. Atlanta, GA, USA: Association for Computing Machinery, Oct. 17–21, 2022, pp. 1471–1480.

[25] H. Chen and P. N. Garner, "Bayesian parameter-efficient fine-tuning for overcoming catastrophic forgetting," *IEEE/ACM Transactions on Audio, Speech, and Language Processing*, vol. 32, pp. 4253–4262, 2024.

[26] Y. Zhai *et al.*, "Investigating the catastrophic forgetting in multimodal large language model fine-tuning," in *Conference on Parsimony and Learning*. Hongkong, China: PMLR, Jan. 3–6, 2024, pp. 202–227.

[27] C. H. Tu *et al.*, "Holistic transfer: Towards non-disruptive fine-tuning with partial target data," *Advances in Neural Information Processing Systems*, vol. 36, pp. 29 149–29 173, 2023.

[28] M. Wortsman *et al.*, "Robust fine-tuning of zero-shot models," in *2022 IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR)*. New Orleans, LA, USA: IEEE, June 18–24, 2022, pp. 7949–7961.

[29] M. Y. Kim, J. Rabelo, K. Okeke, and R. Goebel, "Legal information retrieval and entailment based on BM25, transformer and semantic thesaurus methods," *The Review of Socionetwork Strategies*, vol. 16, pp. 157–174, 2022.

[30] R. Ramadita *et al.*, "Sequran: Sentence-BERT for specific domain of Quran search engine," 2024, presented in International Invention Competition for Young Moslem Scientists 2024.

[31] X. Zhang *et al.*, "MIRACL: A multilingual retrieval dataset covering 18 diverse languages," *Transactions of the Association for Computational Linguistics*, vol. 11, pp. 1114–1131, 2023.

[32] X. Zhang, X. Ma, P. Shi, and J. Lin, "Mr. TyDi:

A multi-lingual benchmark for dense retrieval," in *Proceedings of the 1st Workshop on Multilingual Representation Learning*. Punta Cana, Dominican Republic: Association for Computational Linguistics, Nov. 2021, pp. 127–137.

[33] J. H. Clark *et al.*, "Tydi QA: A benchmark for information-seeking question answering in ty pologically di verse languages," *Transactions of the Association for Computational Linguistics*, vol. 8, pp. 454–470, 2020.

[34] Sequran, "Indo-Islamic queries and answers dataset," 2025. [Online]. Available: https://huggingface.co/datasets/ramadita/Indo-Islamic-QA

[35] A. Pauli, L. Derczynski, and I. Assent, "Anchoring fine-tuning of sentence transformer with semantic label information for efficient truly few-shot classification," in *Proceedings of the 2023 Conference on Empirical Methods in Natural Language Processing*. Singapore: Association for Computational Linguistics, Dec. 2023, pp. 11 254–11 264.

[36] T. Sultana, A. K. Mandal, H. Saha, M. N. Sultan, and M. D. Hossain, "Intent identification by semantically analyzing the search query," *Modelling*, vol. 5, no. 1, pp. 292–314, 2024.

[37] N. L. P. I. Candrawengi and M. W. P. Dananjaya, "Machine learning approaches for search intent-driven website optimization," in *2024 10th International Conference on Smart Computing and Communication (ICSCC)*. Bali, Indonesia: IEEE, July 25–27, 2024, pp. 197–202.

[38] C. D. Schultz, "Informational, transactional, and navigational need of information: Relevance of search intention in search engine advertising," *Information Retrieval Journal*, vol. 23, no. 2, pp. 117–135, 2020.

[39] F. Rahman, *Tema-tema pokok Al-Quran*. Al Mizan, 2018.

[40] C. P. Harshitha and N. R. Sunitha, "Topic identification for semantic grouping based on hidden Markov model," in *2020 5th International Conference on Communication and Electronics Systems (ICCES)*. Coimbatore, India: IEEE, June 10–12, 2020, pp. 932–937.

[41] T. D. Jayasiriwardene and G. U. Ganegoda, "Keyword extraction from Tweets using NLP tools for collecting relevant news," in *2020 International Research Conference on Smart Computing and Systems Engineering (SCSE)*. Colombo, Sri Lanka: IEEE, Sep. 24, 2020, pp. 129–135.

[42] R. Khatun and A. Sarkar, "Deep-KeywordNet: Automated English keyword extraction in documents using deep keyword network based ranking," *Multimedia Tools and Applications*, pp. 68 959–68 991, 2024.

[43] R. Campos, V. Mangaravite, A. Pasquali, A. Jorge, C. Nunes, and A. Jatowt, "YAKE! Keyword extraction from single documents using multiple local features," *Information Sciences*, vol. 509, pp. 257–289, 2020.

[44] A. Gupta, A. Chadha, and V. Tewari, "A natural language processing model on BERT and YAKE technique for keyword extraction on sustainability reports," *IEEE Access*, vol. 12, pp. 7942–7951, 2024.

[45] R. A. Yunmar, A. Setiawan, and H. Tantriawan, "The combination of YAKE and language processing for unsupervised term extraction ontology learning," in *IOP Conference Series: Earth and Environmental Science*, vol. 537. IOP Publishing, 2020.

[46] M. Grootendorst *et al.*, "MaartenGr/KeyBERT: v0.9," 2025. [Online]. Available: https://zenodo.org/records/14831372

[47] J. Sammet and R. Krestel, "Domain-specific keyword extraction using BERT," in *Proceedings of the 4th Conference on Language, Data and Knowledge*. Vienna, Austria: NOVA CLUNL, 2023, pp. 659–665.

[48] B. Issa, M. B. Jasser, H. N. Chua, and M. Hamzah, "A comparative study on embedding models for keyword extraction using KeyBERT method," in *2023 IEEE 13th International Conference on System Engineering and Technology (ICSET)*. Shah Alam, Malaysia: IEEE, Oct. 2, 2023, pp. 40–45.

[49] M. Liu, X. Luo, G. Wang, and W. Z. Lu, "Intelligent information extraction from government on-site inspection reports of construction projects: A graph-based text mining approach," *Advanced Engineering Informatics*, vol. 58, 2023.

[50] K. Liu, Y. Li, Y. Qi, N. Qi, and M. Zhai, "Text information mining in cyberspace: An information extraction method based on T5 and KeyBERT," in *2024 IEEE 9th International Conference on Data Science in Cyberspace (DSC)*. Jinan, China: IEEE, Aug. 23–26, 2024, pp. 621–628.

[51] E. T. Luthfi, Z. I. M. Yusoh, and B. M. Aboobaider, "BERT based named entity recognition for automated Hadith narrator identification," *International Journal of Advanced Computer Science and Applications*, vol. 13, no. 1, pp. 604–611, 2022.

[52] S. L. Chen, P. J. Burns, T. J. Bolt, P. Chaudhuri, and J. P. Dexter, "Leveraging part-of-speech tag-

ging for enhanced stylometry of Latin literature," in *Proceedings of the 1st Workshop on Machine Learning for Ancient Languages (ML4AL 2024)*. Hybrid in Bangkok, Thailand and Online: Association for Computational Linguistics, Aug. 2024, pp. 251–259.

[53] S. Jin, S. Chen, and X. Xie, "Property-based test for part-of-speech tagging tool," in *2021 36th IEEE/ACM International Conference on Automated Software Engineering (ASE)*. Melbourne, Australia: IEEE, Nov. 15–19, 2021, pp. 1306–1311.

[54] M. Nadim, D. Akopian, and A. Matamoros, "A comparative assessment of unsupervised keyword extraction tools," *IEEE Access*, vol. 11, pp. 144 778–144 798, 2023.

[55] T. Taipalus, "Vector database management systems: Fundamental concepts, use-cases, and current challenges," *Cognitive Systems Research*, vol. 85, pp. 1–8, 2024.

[56] J. J. Pan, J. Wang, and G. Li, "Survey of vector database management systems," *The VLDB Journal*, vol. 33, pp. 1591–1615, 2024.

[57] X. H. Lu, "BM25S: Orders of magnitude faster lexical search via eager sparse scoring," 2024. [Online]. Available: https://arxiv.org/abs/2407.03618

[58] D. Teodoro *et al.*, "Information retrieval in an infodemic: The case of COVID-19 publications," *Journal of Medical Internet Research*, vol. 23, no. 9, 2021.

[59] K. Pfleger and B. Larson, "System and method for determining a composite score for categorized search results," 2010.

[60] N. Reimers and I. Gurevych, "The curse of dense low-dimensional information retrieval for large index sizes," in *Proceedings of the 59th Annual Meeting of the Association for Computational Linguistics and the 11th International Joint Conference on Natural Language Processing (Volume 2: Short Papers)*. Online: Association for Computational Linguistics, Aug. 2021, pp. 605–611.

[61] L. C. Chen, "A study of optimizing search engine results through user interaction," *IEEE Access*, vol. 8, pp. 79 024–79 045, 2020.

[62] M. D. Almadhoun and N. H. A. H. Malim, "Effects of using multi-category web pages on rank estimation of Google search engine results page," *Web Intelligence*, vol. 23, no. 1, pp. 39–55, 2025.

[63] A. Urman and M. Makhortykh, "You are how (and where) you search? Comparative analysis of web search behavior using web tracking data," *Journal of Computational Social Science*, vol. 6, pp. 741–756, 2023.

[64] N. Craswell, B. Mitra, E. Yilmaz, D. Campos, E. M. Voorhees, and I. Soboroff, "TREC deep learning track: Reusable test collections in the large data regime," in *Proceedings of the 44th International ACM SIGIR Conference on Research and Development in Information Retrieval*. Association for Computing Machinery, July 11–15, 2021, pp. 2369–2375.

[65] E. Bassani, "ranx: A blazing-fast python library for ranking evaluation and comparison," in *European Conference on Information Retrieval*. Stavanger, Norway: Springer, April 10–11, 2022, pp. 259–264.

[66] Z. H. Amur, Y. K. Hooi, G. M. Soomro, H. Bhanbhro, S. Karyem, and N. Sohu, "Unlocking the potential of keyword extraction: The need for access to high-quality datasets," *Applied Sciences*, vol. 13, no. 12, pp. 1–19, 2023.

[67] M. Wrzalik and D. Krechel, "CoRT: Complementary rankings from transformers," in *Proceedings of the 2021 Conference of the North American Chapter of the Association for Computational Linguistics: Human Language Technologies*. Online: Association for Computational Linguistics, June 2021, pp. 4194–4204.

[68] S. Chang, G. J. Ahn, and S. Park, "Improving performance of neural IR models by using a keyword-extraction-based weak-supervision method," *IEEE Access*, vol. 12, pp. 46 851–46 863, 2024.

[69] G. Deena and K. Raja, "Keyword extraction using latent semantic analysis for question generation," *Journal of Applied Science and Engineering*, vol. 26, no. 4, pp. 501–510, 2022.

[70] B. Plank, "Sliced at SemEval-2022 Task 11: Bigger, better? Massively multilingual LMs for multilingual complex NER on an academic GPU budget," in *Proceedings of the 16th International Workshop on Semantic Evaluation (SemEval-2022)*. Seattle, United States: Association for Computational Linguistics, July 2022, pp. 1494–1500.

[71] L. Gao, Z. Dai, T. Chen, Z. Fan, B. Van Durme, and J. Callan, "Complement lexical retrieval model with semantic residual embeddings," in *European Conference on Information Retrieval*. Virtual: Springer, March 28–April 1, 2021, pp. 146–160.