JKAU: Comp. IT. Sci., **Vol. 12** No. 2, pp: 55 – 66 (2023 A.D.)
DOI: 10.4197/Comp.12-2.5

Fast and Efficient Multilingual Unified MOOCs Semantic Search Engine (UMSSE)

Ahmad Fajar Tatang, Abdullah M. Algarni
Department of Computer Science
Faculty of Computing and Information Technology, King Abdulaziz University
Jeddah, Saudi Arabia
atatang@stu.kau.edu.sa, amsalgani@kau.edu.sa

***Abstract*—Massive Open Online Courses (MOOCs) proliferation has created a demand for effective search engines to help learners identify and enrol in courses that meet their needs. However, building a multilingual unified MOOC search engine that can provide comprehensive search results has been challenging due to the many platforms in different languages and the diversity of available courses. This paper proposed and implemented a model for Unified MOOCs Semantic Search Engine (UMSSE). This UMSSE leveraged a combination of text encoder models and an Approximate Nearest Neighbor (ANN) algorithm to improve the speed and accuracy of search results. The model integrated data from the platform and multiple languages, utilizing Natural Language Processing (NLP) and machine learning techniques to understand the meaning of search queries and recommend relevant courses. The performance of the proposed model was evaluated using a dataset of various MOOC course descriptions. The results showed that it outperformed traditional keyword-based search engines regarding performance metrics. Several applied examples further illustrated how the proposed model could improve the speed and effectiveness of MOOCs search and recommend appropriate courses to users.**

Keywords—Semantic Search Engine, Metadata, Natural Language Processing, Unified MOOCs, Information Retrieval Techniques

I. INTRODUCTION

The recent rapid growth of Massive Open Online Courses (MOOCs) has transformed the traditional delivery and accessibility of education. The platform provides unlimited access to educational materials for individuals interested in taking a course without requiring physical attendance. Furthermore, these online courses provide learners with a flexible and convenient means of accessing high-quality educational content from renowned institutions worldwide [1]. They have emerged as a preferred choice for learners seeking to upskill or reskill and for educators aiming to expand their reach to a broader audience.

There are several platforms, including MOOCs providers [2]–[4], universities [5], institutes [6], [7], and even individuals, that offer courses to the public. However, with the growing number of MOOCs available, it has become increasingly challenging for learners to locate and access the most relevant and valuable courses that meet their needs. Traditional search engines utilized in this regard only rely on keyword matching, which can be limited in their ability to understand the context and intent of a search query. This deficiency can lead to a suboptimal user experience for learners as they have to sift through numerous irrelevant or unrelated search results before finding the courses of interest.

To address this limitation, a model for a Unified MOOCs Semantic Search Engine (UMSSE) was proposed to aid learners in discovering and accessing multiple MOOCs based on the meaning and context of their search queries. Semantic technologies allow a UMSSE to understand the intent and context of a search query, known as a semantic search engine, and provide more relevant and accurate search results. This can significantly enhance the user experience for learners and assist in finding the most suitable platform for their needs.

The objective of this research is to explore the benefits and challenges of implementing the UMSSE that utilizes approximate nearest neighbor (ANN) algorithms and Sentence Bidirectional Encoder Representations from Transformers (S-BERT) comprising Bi-Encoder and Cross-Encoder, in order to improve the accuracy and precision of search results. A platform was developed to verify and evaluate the proposed model's search accuracy and latency performance. In addition to the above, the potential applications and future directions of UMSSEs are also examined.

The UMSSE has significant benefits for both learners and MOOC providers. It can facilitate finding and accessing relevant and high-quality MOOCs for learners, improving their learning experience and outcomes. For MOOCs providers, it can increase the visibility and accessibility of their courses, potentially leading to higher enrollment and revenue.

The main contribution of this paper is the development of a novel semantic search engine for MOOCs, which employs advanced NLP techniques to improve the accuracy and relevance of search results based on the meaning of user queries within a short time. The rest of this paper is organized as follows, Section 2 reviews the literature and related work on unified MOOCs and the retrieving techniques. Section 3 describes the proposed model; meanwhile, the implementation and the evaluation of the results are discussed in Section 4. Furthermore, Sections 5 and 6 describe the analysis of the results and the limitations and challenges faced in this research, respectively. The paper concludes with a summary of the main contributions and suggestions for future work.

II. Background

A. *Information Retrieval Techniques*

Information retrieval (IR) techniques are methods and approaches to search and retrieve relevant information from extensive data collections, such as databases, the web, or other digital media [8]. Some classical and modern IR techniques include keyword matching, Boolean search, Latent semantic indexing, Latent Dirichlet allocation, and Semantic search.

Keyword matching involves searching for documents containing specific sets of keywords. Although this technique is simple and widely used, it needs to be improved in understanding the context and intent of a search query [8], [9].

The boolean search uses logical operators, such as AND, OR, and NOT, to combine keywords and narrow the search results. This technique is more effective at finding relevant documents but still needs to be improved in its ability to understand the context and intent of a search query [10].

Latent semantic indexing (LSI) uses mathematical techniques to extract the latent (hidden) semantic relationships between terms and documents. This improves the accuracy of search results by considering the relationships between terms rather than just the presence of specific keywords [11]. Latent Dirichlet allocation (LDA) is a probabilistic model that can identify the topics present in a document and represent them as a distribution of topics. This method can identify the topics in a search query and find documents that match those topics [12], [13].

Semantic search involves using NLP and semantic technologies to understand the meaning and context of a search query and provide more relevant and accurate search results. This technique is more effective at finding relevant documents but requires a more complex and computationally intensive implementation [14].

Apart from the traditional information retrieval techniques, modern approaches have been widely developed and have yielded better results. For example, Nimmani et al. [15] proposed combining multiple IR techniques with change impact analysis (CIA) and a Bag of Words to identify the potential consequences of a replacement or manage necessary changes to achieve a desired outcome. In the research, a neural network-based Long Short-Term Memory Recurrent Neural Network (LSTM-RNN) algorithm is proposed to find similar documents, and the RMSprop optimization model is used to improve the learning rate and precision. The experimental results show that the proposed method has better accuracy than others.

Moon Soo Cha et al. [16] employed Latent Dirichlet Allocation (LDA) to retrieve the information from a content-based document system, while Qiu et al. [17] introduced an unsupervised product quantization model for document retrieval. Hui et al. [18] also proposed a new model for neural information retrieval. Consequently, these modern approaches have improved accuracy and precision compared to traditional methods.

B. *Information Retrieval in MOOCs*

In MOOCs, the large volume of educational materials can challenge learners to locate and access relevant content. To address this issue, various information retrieval techniques have been proposed, one of which is using phrase recommendation systems. Baeza-Yates & Ribeiro-Neto [19] stated that recommendation and IR systems are similar in their retrieval and presentation of information to users.

A widely used approach in MOOCs is NLP, which enables automatic analysis of text data, including course descriptions, lectures, and other materials, to extract useful information and facilitate search and retrieval. For example, Sakboonyarat and Tantatsanawong [20] proposed a course recommendation using deep learning with a multilayer perceptron architecture. Tan et al. [21] utilized an autoencoder to recommend courses, while Ma, H. et al. [22] combined Latent Semantic Indexing (LSI) and Document to Vector (D2V) for course recommendation.

Machine learning is another approach used to analyze MOOC data, such as student interactions and course performance, to identify patterns and relationships that can improve the search and retrieval of online course materials. Zhang H. et al. [23] introduced a personalized course recommendation system utilizing a deep belief network (DBN) for the first time. They combined accurate user data with traditional recommendation methods and trained the developed model, called DBNCF. Through experiments and comparative analysis, it was observed that DBNCF had a strong performance in pre-diction classification and high recommendation efficiency.

Data mining is a technique widely used in MOOCs to improve the search and retrieval of educational materials. This involves analyzing large datasets using the algorithm to identify patterns and trends that can improve the search and retrieval of MOOC materials. Several studies [[24]–[28] have used data mining to explore MOOCs.

C. *Information Retrieval in Unified MOOCs*

A unified MOOCs platform is a centralized location aggregating courses in several languages, enabling users to access materials from various sources without visiting the website. This approach offers users a convenient and seamless way to discover and enrol in the platform from multiple providers within a single interface. It also improves the user experience, making finding and enrolling in courses matching their interests and needs easier.

Courserush, a MOOCs search engine, was proposed by Lee [28]. It uses manual data scraping and the BM25 ranking algorithm to search for courses on three specific platforms: edX, Udemy, and Coursera. However, the use of manual data collection and indexing caused several limitations. In order to overcome these constraints, an improved algorithm was created that employs the BM25 ranking mechanism alongside a custom-built ranking function based on Apache Lucene. Furthermore, popular platforms like Coursera, Udemy, and edX were scraped to populate the index with relevant data. Adopting these innovative methods has substantially improved the efficacy and precision of the Courducate search engine, rendering it an up-and-coming tool for searching MOOCs [26].

Using a Matrix Factorization model, MoocRec.com adopts a hybrid approach that combines a search engine and a content recommender system. The system solely sources data from edX and Coursera [27]. Kagemann and Bansal, on the other hand, created Linked Data by consolidating data from multiple MOOC providers using their ontology. They incorporated this data into a web application that enables users to explore courses from different MOOC providers [25]. Similarly, Alzahrani and Meccawy [24] proposed a hybrid search engine and recommender system model that helps users browse courses on a single platform. However, the approach is limited to vertical search with clustering components for recommendation functionality.

Despite the potential benefits of a unified MOOCs platform, relatively limited research has been conducted on this topic. Aside from the challenges of aggregating data from multiple course providers, creating a coherent and structured presentation of the aggregated data is critical in developing a unified MOOCs platform. Therefore, this paper aims to address these difficulties and maximize performance using various techniques.

D. *Search Engine vs Semantic Search Engine*

A search engine is a software program that helps users search for information on the Internet using keywords or phrases the user enters. This is the most common type of search that people are familiar with and is used by popular web search engines. When a user enters a few keywords, the search engine returns a list of documents that match those keywords or their variations.

On the other hand, a semantic search engine uses NLP and ML algorithms to analyze the context and intent behind a query and delivers more accurate and relevant results. This type of search engine goes beyond traditional keyword-based searches, attempting to understand the meaning of the query rather than just matching keywords.

Some research has been conducted to create a semantic search engine using a different approach. For example, Jiang S. et al. [29] proposed a hybrid indexing approach that computes semantic similarity based on lexical vocabularies and adjusts scores based on the known relatedness of concepts defined by an ontology model. Similarly, Pan, Z [30] optimized a retrieval system algorithm using a new algorithm based on a semantic search engine in a digital library application, resolving the sorting problem of the retrieval result.

An et al. [1] introduced a virtual search engine for retrieving MOOC courses in Chinese based on user keywords. The search engine uses a semantic approach, which involves comprehending user intent, query context, and word relationships to produce the most accurate results possible. Fazzinga and Lukasiewicz provided a detailed explanation of this domain [31].

E. *Approximate Nearest Neighbor*

Approximate nearest neighbor (ANN) search is a powerful technique for retrieving the nearest neighbors of a given query point from a large dataset. This algorithm is particularly advantageous for handling high-dimensional data, as it can achieve excellent search performance while significantly reducing computational cost compared to the traditional approach.

One popular library for implementing ANN search is Facebook AI Similarity Search (FAISS). It is a robust C++ library with Python bindings that offers a range of ANN algorithms, including k-means, hierarchical clustering, and quantization-based methods [32].

Several investigations have been performed regarding the effectiveness of FAISS for ANN search. For instance, the research by [33] compared the performance of FAISS with other ANN libraries on various tasks such as image retrieval and recommendation systems. The results showed that FAISS outperformed other libraries on a range of datasets and use cases, making it an ideal choice for real-world applications.

Qin, C. et al. [34] evaluated the performance of FAISS on big spectral data and found that it achieved a high accuracy rate and demonstrated good scalability. Gupta, A. et al. [35] also evaluated the effectiveness of different features for image retrieval extracted from a Convolutional Neural Network (ConvNet) and 3-D histograms in the HSV colour space and the combination of these two features. Although there was a slight reduction in accuracy, FAISS significantly reduced retrieval time, making it a robust and scalable tool for image retrieval applications. It is important to note that this approach has yet to be tested on massive datasets. Previous research has also proven that FAISS improved the quality of results, making it a suitable choice for large-scale ANN search tasks.

F. *Pre-Trained Text Encoder Model*

A pre-trained text encoder is a deep learning model trained on vast textual data using unsupervised learning techniques. It learns general-purpose representations of text that can be fine-tuned for downstream NLP tasks. It is crucial to note that this approach has gained popularity as it has proven to improve the performance of NLP tasks, such as text classification, sentiment analysis, and language translation. An example of a widely-used pre-trained text encoder is Bidirectional Encoder Representations from Transformers (BERT), introduced by [36], which serves as a starting point for many downstream NLP tasks.

Individuals can save time and resources using a pre-trained text encoder since they are not required to train a deep-learning model from scratch. Instead, the user can fine-tune the pre-trained model for specific tasks, reducing the time and resources needed to achieve satisfactory results.

For example, BERT is commonly used in semantic search as it sets a new standard of excellence in various sentence classification and phrase-pair regression challenges. This model utilizes a cross-encoder approach in which two phrases are input to a transformer network, and the target value is predicted. However, this approach is only optimal for some pair regression tasks due to the numerous possible combinations. To address this issue, the research by [37] introduces Sentence BERT (S-BERT), a variant of the pre-trained BERT network that uses siamese and triplet network architectures to generate semantically significant phrase embeddings. Notably, these embeddings can be compared using cosine similarity.

III. Proposed Research

A. *Dataset and Methods*

In this research, the course data was collected in two ways: from the public API and crawling the website using a particular Python program. Not all MOOC providers offered an API to retrieve data, which required the creation of a custom scraper bot for web crawling. A bot was designed to ensure the reliability of the scraper bot and avoid blocking by the website. Most sources in the dataset were obtained from classcentral.com, which has gathered several providers into one place. The comprehensive dataset included 73,409 courses in 75 languages from 65 MOOC sources. The bot was run on a machine with an Intel(R) Xeon(R) CPU @ 2.30GHz with four cores, 26G RAM, 167G disk space, and a Tesla T4 GPU model. The dataset consisted of course name, description, institute, provider, instructor, language, rating, and link. This dataset became the most comprehensive data on MOOCs and continues to grow.

The data presented in Table 1 shows that Udemy has the highest number of courses on the list. This finding is significant because it highlights the popularity and reach of Udemy as a platform for online education. Additionally, Table 2 describes the language instructions for the courses in the dataset, with English, Spanish, and Portuguese being the most commonly used languages. These results are essential for understanding the linguistic diversity of MOOCs and the potential audience for these courses. The dataset and source code are available upon request for the replication and extension of this research.

TABLE I. List of Course Providers

Course Providers	**Total Courses**
Udemy	21370
Youtube Learning	9274
Linkedin Learning	7912
Coursera	5833
Pluralsight	5318
Study.com	3411
Skillshare	3196
AWS Skill Builder	3036
Domestika	2345
CreativeLive	1725
FutureLearn	1429
edX	1401
Independent	922
OpenLearn	805
others	5432

TABLE II. List of Used Language in the Course

Languages	**Total Courses**
English	59973
Spanish	4689
Portuguese	1591
French	1178
Turkish	900
Arabic	851
Hindi	712
Japanese	693
German	622
Chinese	583
Italian	372
Korean	323
Thai	308
Indonesian	281
others	846

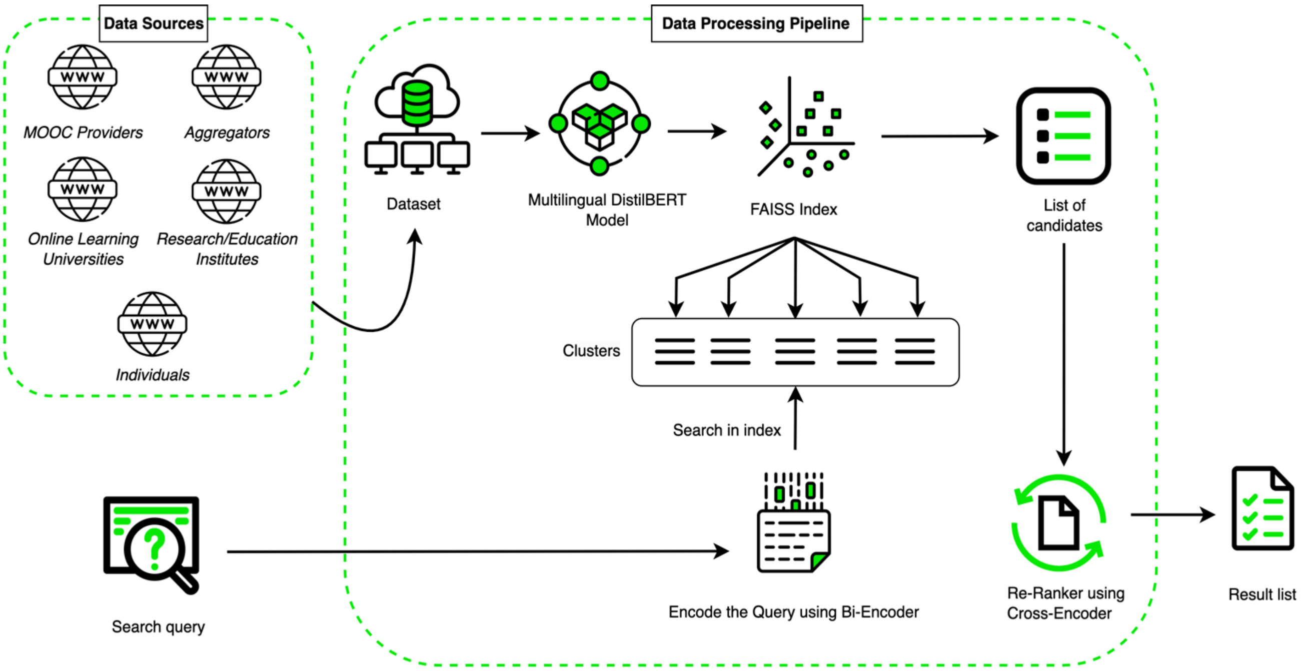


Fig. 1. Overall flow of the proposed UMSSE Model

B. *Proposed Model*

The proposed UMSSE model aimed to enable users to browse and retrieve relevant courses on a single platform, using keyword meaning. The model-building process, as shown in Fig. 1, began with the collection of data from several sources.

The dataset from scraping MOOCs was cleaned and prepared in the data processing pipeline before being transformed by a multilanguage DistilBERT model. This is a smaller and faster version of the BERT used for NLP [38], introduced in 2019 by Hugging Face and Microsoft. A model that was 40% smaller, faster, and required less memory to run was developed through the implementation of a technique called distillation. Despite its reduced size, DistilBERT performs comparably to BERT on many NLP tasks, making it a popular choice for applications with limited computational resources. Since the model utilized the same transformer architecture as BERT, it was well-suited for various NLP tasks.

In the following process, Asymmetric Semantic Search was utilized. This search technique often employed a brief query, such as a question or a few keywords, and sought a more comprehensive text that addressed the inquiry. For example, a question like "What is Artificial Intelligence?" could be used. The desired answer could be, "Artificial Intelligence is the replication of human cognitive processes by machines, primarily computer systems." In asymmetric jobs, flipping the query and corpus items often needed to be clarified, as this technique typically relies on a specific search direction.

Asymmetric Semantic Search was equipped with distilbert-base-multilingual-cased for handling multiple languages. The dataset was divided into buckets to accelerate the search process, and only a subset of buckets (nprobe buckets) was accessed during search time. Clustering was conducted on a representative dataset vector sample, typically sampled from the dataset. The recommended sample size for this sample was provided, and the index used was the IndexIVFFlat, which was quantized with Index-FlatL2.

Through machine learning, the semantic search captured the context, data, and search queries. Generally, machine learning models made a trade-off between high accuracy and speed. As the precision of a model increased, it became more computationally expensive. When discussing search or Information Retrieval, accurate search results were expected to cover a wide range of queries while maintaining a high recall speed.

During the search or semantic sentence matching process, there is often a trade-off between Bi-Encoder and Cross-Encoder models. Bi-Encoder models were faster but less accurate, while CrossEncoder was slower but more accurate. Both models could be utilized in a search pipeline to maximize their usefulness. Algorithm 1 explained how search queries were processed, of which Q represented a query input, and I denoted a document index as function parameters. Bi-Encoder performed vectorization of the input, which was then stored as a query vector (QV). The document index was searched based on the vector, and the first result or candidates (K) were returned from the index with a total of N. Furthermore, C indicated the result based on the Bi-Encoder. The documents were validated by comparing each score and returned in ascending order using Cross-Encoder (CS).

Utilizing an unsupervised method for Encoder fine-tuning produced more accurate results. With a query and relevant passage information, Bi-Encoder could efficiently fine-tune a sentence-transformer model for the dataset. This research did not focus on the pre-training technique of the transformer model; hence, synthetic query generation was utilized. The unsupervised model was fine-tuned on MS MARCO (MAchine Reading COmprehension), a pre-trained model designed explicitly for passage ranking using the MS MARCO dataset [39]. The model generated high-quality sentence embedding and was used for various NLP tasks, including text classification and question answering. It was proven to outperform other pre-trained models and was widely recognized as a benchmark in the field. Subsequently, it was evaluated on a collection of zero-shot search benchmark tasks from BEIR [40]. This method enabled the creation of a model for asymmetric semantic search without requiring training data.

Through implementing BEIR, each chunk was designed to include no more than five synthetically generated queries. The information in a paragraph was presented as questions, and this knowledge tuple was utilized to fine-tune an S-BERT model that captured the semantic and syntactic information mapping between these tuples.

Algorithm 1 Re-Ranker Result Documents

```
procedure search_courses(Q, I)
    QV ← bi encodes Q
    K ← I(Q, N)
    C ← Q in N
    CS ← cross-encoder score
    for S in CS do
        result ← S
    end for
    return result ASC
end procedure
```

IV. Results

UMSSE was developed using various open-source tools or libraries, including Python (programming language), FAISS (ANN algorithms), and TensorFlow (bi-encoder and cross-encoder models). The search engine was built upon a multilingual DistilBERT model, which was trained using a dataset of MOOCs courses descriptions in multiple languages. In addition, customized code was created for data scraping and pre-processing. To use the system, it is crucial to define the XML file or RSS feed of the MOOC provider or platform and map the scrapped data to the appropriate field in the database. The list of providers and languages that were successfully obtained can be seen in Tables 1 and 2.

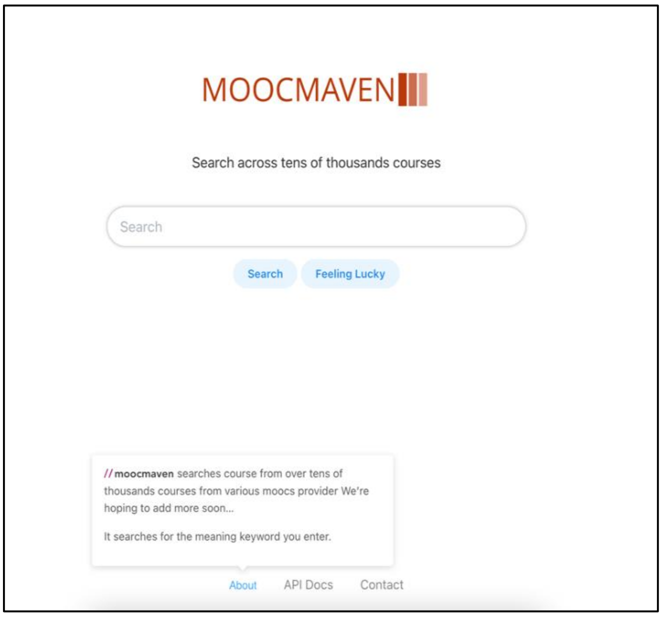
UMSSE was trained using a dataset of MOOCs annotated with semantic tags, descriptions, and other supporting data such as course name, instructor, and institute. Various metrics, including search accuracy and recall, were used to evaluate the UMSSE's performance. However, the course title and description are the primary data source used for training and evaluation.

The selection of languages in this research was based on the top languages in the course database. Additionally, personal preference played a role in including Arabic, Indonesian, and German. Recall, precision and F1 score is used as evaluation metric to measure the effectiveness of the search engine in retrieving relevant documents from a collection. It was computed as the number of relevant documents retrieved by the search engine divided by the total number of relevant documents in the collection [41]. Table 3 displays the search engine's performance for various languages and hit values. The analysis of the retrieval performance across multiple languages indicates the overall effective performance of the system in retrieving relevant documents. The precision values are consistently perfect at 100.00% for all languages, indicating that all retrieved documents are relevant. The recall values vary across languages, with some languages achieving higher recall rates than others.

The F1 scores, which consider precision and recall, indicate a good balance between the two measures for most languages. The F1 scores range from 69.57% to 100.00%, with most languages achieving F1 scores above 80.00%. This suggests that the retrieval system successfully captures a substantial number of relevant documents while maintaining high precision.

The retrieval time represents the speed at which the model can fetch relevant information for a given query. Among the languages tested, the search engine demonstrated relatively fast retrieval times for Chinese (0.00565ms), Portuguese (0.00562ms), Indonesian (0.00657ms), Arabic (0.00710ms), English (0.00553ms), and French (0.00515ms). However, it exhibited a comparatively slower retrieval time for Spanish (0.03232ms) and German (0.02913ms).

Overall, the results demonstrate the effectiveness of the retrieval system across different languages, with solid precision and balanced performance in terms of recall and F1 score. The use of a multilingual DistilBERT model as the base for the search engine appeared to be effective at capturing the context and semantics of the data and search queries. The sample result is seen in Fig. 3. This advanced semantic search engine was tailored exclusively for MOOCs and employed cutting-edge algorithms to provide unparalleled search results. The MOOCMaven is a platform for academics and lifelong learners seeking to identify the most informative and rigorous online courses available. Its sophisticated search capabilities allowed users to easily navigate through vast repositories of MOOCs and access the most relevant based on the query. This platform can be accessed through the link [42].



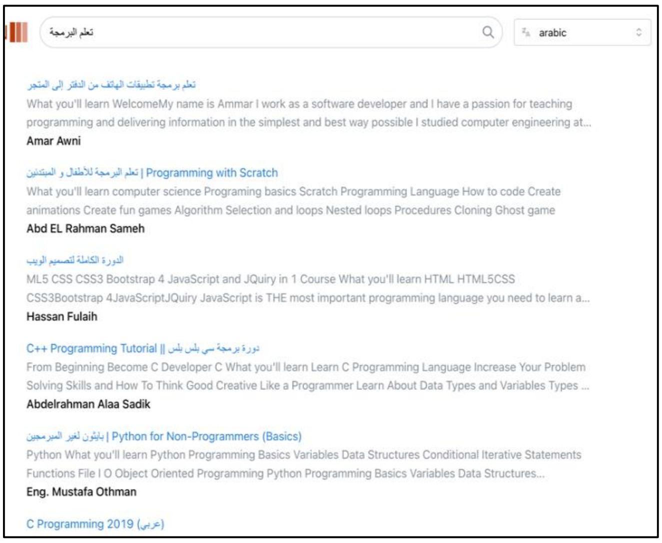
Search across tens of thousands courses

// moocmaven searches course from over tens of thousands courses from various moocs provider We're hoping to add more soon...

It searches for the meaning keyword you enter.

Fig. 2. MoocMaven.com platform Interface

Fig. 3. Sample result in Arabic with the query تعلم البرمجة (learn programming)



تعلم برمجة تطبيقات الهاتف من الدفتر إلى المتجر
What you'll learn WelcomeMy name is Ammar I work as a software developer and I have a passion for teaching programming and delivering information in the simplest and best way possible I studied computer engineering at...
Amar Awni

تعلم البرمجة للأطفال و المبتدئين | Programming with Scratch
What you'll learn computer science Programing basics Scratch Programming Language How to code Create animations Create fun games Algorithm Selection and loops Nested loops Procedures Cloning Ghost game
Abd EL Rahman Sameh

الدورة الكاملة لتصميم الويب
ML5 CSS CSS3 Bootstrap 4 JavaScript and JQuiry in 1 Course What you'll learn HTML HTML5CSS CSS3Bootstrap 4JavaScriptJQuiry JavaScript is THE most important programming language you need to learn a...
Hassan Fulaih

C++ Programming Tutorial || دورة برمجة سي بلس بلس
From Beginning Become C Developer C What you'll learn Learn C Programming Language Increase Your Problem Solving Skills and How To Think Good Creative Like a Programmer Learn About Data Types and Variables Types ...
Abdelrahman Alaa Sadik

بايثون لغير المبرمجين | Python for Non-Programmers (Basics)
Python What you'll learn Python Programming Basics Variables Data Structures Conditional Iterative Statements Functions File I O Object Oriented Programming Python Programming Basics Variables Data Structures...
Eng. Mustafa Othman

C Programming 2019 (عربى)

V. Discussion

The proposed model (UMSSE) has the potential to effectively perform semantic search tasks, particularly in a multilingual context, while also being efficient and flexible. Based on its advanced techniques, UMSSE is able to understand the intent and context of a search query and provide more relevant and accurate search results. It also delivers lightning-fast and highly efficient outcomes across multiple languages, making it an indispensable model for anyone seeking to navigate the vast and complex world of MOOCs. This significantly

TABLE III. COMPARISON OF THE UMSSE MODEL WITH OTHER MOOCS RETRIEVING MODELS

Name	Total of MOOCs	Total Used Languages	Used Techniques	Features	Main Results
MOOCs Recommender Search Engine	8	2	TF-IDF	Closed data, search, recommendation	The system can retrieve courses in both Arabic and English languages courses.
MoocRec	2	1	Content (Video/Topic Modeling) Recommender System	Closed data, search, recommendation	Success presents the selected courses based on questionnaires to identify his learning styles.
Courducate	5	1	BM25 Ranking	Closed data, search, filtering	MOOCs course search engine based on user filter
Courserush	5	1	BM25 Ranking	Closed data, search	MOOCs search engine
MoocLink	3	1	Query Expansion and Classification	Closed data, search	A method for collecting and generating Linked Data from three MOOCs Providers were presented
Chinese MOOCs Search Engine	16	1	Custom Course Rank	Closed data, search	MOOCs search engine in Chinese
UMSSE	65	75	ANN, Bi-Encoder, Cross-Encoder	Open data, Semantic Search	The system can deliver courses based on the meaning of user query

improved the user experience for learners and helped them locate the most appropriate platform required. Consequently, the system is more scalable for long-term use.

UMSSE has several potential applications and future directions, one of which is in the field of education, where it can enhance the accessibility and effectiveness of MOOCs for learners. The model can also be extended to other domains, such as e-commerce, city finder, or any other system that requires a search engine. In these domains, it can provide more relevant and accurate search results based on the meaning and context of a search query.

Aside from providing courses that matched the search keyword, UMSSE was able to understand its meaning. This led to more accurate search results than currently available systems [43]–[47]. Specifically, the system supported searching in eight languages of instruction for MOOCs, making it more effective for non-English queries.

Several systems suffered from uncovered languages, as they only returned results based on keywords, which led to relevant courses not appearing in the search results. The retrieval mechanism also obtained irrelevant documents; hence, a re-ranker based on a cross-encoder was utilized to evaluate the relevance of all candidates for the current search query. In addition, the indexing dataset using FAISS provided excellent speed and data parity, making the system excel compared to others. This is proven by the comparisons in Table 3, which explain the comparisons with other proposed systems.

VI. CONCLUSIONS AND FUTURE WORK

Semantic search is gaining significant attention from both academia and the business community. Although research prototypes are still being developed, early research-based products are already commercially available. This paper provides an overview of the current state-of-the-art semantic search, combining fine-tuned S-BERT for the unified MOOCs.

The results are very satisfying, indicating the low latency of the final list with high recall. It implies that the system effectively retrieves relevant information for user queries, reducing the time required to search for the target course. Furthermore, the precision and F1 score results complement the overall performance evaluation, indicating the accuracy and reliability of the system in providing precise matches. These findings demonstrate the successful application of the methodology in optimizing the search process and improving user experience.

There are several possibilities for future research from this point. An example is enabling the system to auto-detect the languages of a query and add a personalized semantic search engine based on the activity log and user information. This system could be further enhanced by supporting additional languages or increasing the number of indexed MOOCs. Achieving this could involve developing continuous data scraping and pre-processing techniques or integrating with existing platforms through API agreements.

ACKNOWLEDGMENT

This work was funded by Institutional Fund Projects under grant no. (IFPIP:986-611-1443). The authors gratefully acknowledge technical and financial support provided by the Ministry of Education and King Abdulaziz University, DSR, Jeddah, Saudi Arabia.

REFERENCES

[1] B. An, T. Qu, H. Qi, and T. Qu, "Chinese MOOC Search Engine," in *Intelligent Computation in Big Data Era*, pp. 453–458. doi: https://doi.org/10.1007/978-3-662-46248-5_55.

[2] "Coursera." https//www.coursera.org (accessed Nov. 10, 2022).

[3] "Edx." https://www.edx.org (accessed Nov. 10, 2022).

[4] "Udemy." https://www.udemy.com (accessed Nov. 10, 2022).

[5] "MIT OpenCourseWare." https://ocw.mit.edu (accessed Nov. 10, 2022).

[6] "Google Digital Garage." https://learndigital.withgoogle.com/digitalgarage/courses (accessed Nov. 10, 2022).

[7] "British Council." https://learnenglish.britishcouncil.org/ (accessed Nov. 10, 2022).

[8] S. Büttcher, C. L. A. Clarke, and G. V. Cormack, *Information retrieval: implementing and evaluating search engines*. Cambridge, Mass: MIT Press, 2010.

[9] G. Salton and C. Buckley, "Term-weighting approaches in automatic text retrieval," *Information Processing & Management*, vol. 24, no. 5, pp. 513–523, Jan. 1988, doi: 10.1016/0306-4573(88)90021-0.

[10] G. Salton and M. J. McGill, *Introduction to modern information retrieval*. in McGraw-Hill computer science series. New York: McGraw-Hill, 1983.

[11] S. Deerwester, S. T. Dumais, G. W. Furnas, T. K. Landauer, and R. Harshman, "Indexing by latent semantic analysis," *J. Am. Soc. Inf. Sci.*, vol. 41, no. 6, pp. 391–407, Sep. 1990, doi: 10.1002/(SICI)1097-4571(199009)41:6<391::AID-ASI1>3.0.CO;2-9.

[12] M. Steyvers, P. Smyth, M. Rosen-Zvi, and T. Griffiths, "Probabilistic author-topic models for information discovery," in *Proceedings of the tenth ACM SIGKDD international conference on Knowledge discovery and data mining*, Seattle WA USA: ACM, Aug. 2004, pp. 306–315. doi: 10.1145/1014052.1014087.

[13] D. M. Blei, A. Y. Ng, and M. I. Jordan, "Latent Dirichlet Allocation," *The Journal of Machine Learning Research*, vol. 3, pp. 993–1022, Mar. 2003.

[14] R. Mihalcea and P. Tarau, "TextRank: Bringing Order into Texts".

[15] P. Nimmani, S. Vodithala, and V. Polepally, "Neural Network Based Integrated Model for Information Retrieval," in *2021 5th International Conference on Intelligent Computing and Control Systems (ICICCS)*, Madurai, India: IEEE, May 2021, pp. 1286–1289. doi: 10.1109/ICICCS51141.2021.9432241.

[16] Moon Soo Cha, So Yeon Kim, Jae Hee Ha, Min-June Lee, Young-June Choi, and Kyung-Ah Sohn, "CBDIR: Fast and effective content based document Information Retrieval system," in *2015 IEEE/ACIS 14th International Conference on Computer and Information Science (ICIS)*, Las Vegas, NV, USA: IEEE, Jun. 2015, pp. 203–208. doi: 10.1109/ICIS.2015.7166594.

[17] Z. Qiu, Q. Su, J. Yu, and S. Si, "Efficient Document Retrieval by End-to-End Refining and Quantizing BERT Embedding with Contrastive Product Quantization." arXiv, Oct. 31, 2022. Accessed: Mar. 21, 2023. [Online]. Available: http://arxiv.org/abs/2210.17170

[18] K. Hui, A. Yates, K. Berberich, and G. de Melo, "Co-PACRR: A Context-Aware Neural IR Model for Ad-hoc Retrieval." arXiv, Nov. 28, 2017. Accessed: Mar. 21, 2023. [Online]. Available: http://arxiv.org/abs/1706.10192

[19] R. Baeza-Yates and B. Ribeiro-Neto, *Modern Information Retrieval: The Concepts and Technology Behind Search 2nd Edition*, 2nd ed. Addison-Wesley Professional, 2011.

[20] S. Sakboonyarat and P. Tantatsanawong, "Massive Open Online Courses (MOOCs) Recommendation Modeling using Deep Learning," in *2019 23rd International Computer Science and Engineering Conference (ICSEC)*, Phuket, Thailand: IEEE, Oct. 2019, pp. 275–280. doi: 10.1109/ICSEC47112.2019.8974770.

[21] J. Tan, L. Chang, T. Liu, and X. Zhao, "Attentional Autoencoder for Course Recommendation in MOOC with Course Relevance," in *2020 International Conference on Cyber-Enabled Distributed Computing and Knowledge Discovery (CyberC)*, Chongqing, China: IEEE, Oct. 2020, pp. 190–196. doi: 10.1109/CyberC49757.2020.00038.

[22] H. Ma, X. Wang, J. Hou, and Y. Lu, "Course recommendation based on semantic similarity analysis," in *2017 3rd IEEE International Conference on Control Science and Systems Engineering (ICCSSE)*, Beijing, China: IEEE, Aug. 2017, pp. 638–641. doi: 10.1109/CCSSE.2017.8088011.

[23] H. Zhang, H. Yang, T. Huang, and G. Zhan, "DBNCF: Personalized Courses Recommendation System Based on DBN in MOOC Environment," in *2017 International Symposium on Educational Technology (ISET)*, Hong Kong: IEEE, Jun. 2017, pp. 106–108. doi: 10.1109/ISET.2017.33.

[24] K. M. Alzahrani and M. Meccawy, "MOOCs One-Stop Shop: A Realization of a Unified MOOCs Search Engine," *IEEE Access*, vol. 9, pp. 160175–160185, 2021, doi: 10.1109/ACCESS.2021.3130841.

[25] S. Kagemann and S. Bansal, "MOOCLink: Building and utilizing linked data from Massive Open Online Courses," in *Proceedings of the 2015 IEEE 9th International Conference on Semantic Computing (IEEE ICSC 2015)*, Anaheim, CA, USA: IEEE, Feb. 2015, pp. 373–380. doi: 10.1109/ICOSC.2015.7050836.

[26] Q. Cheng and Y. Gao, "Courducate -- An MOOC Search and Recommendation System," p. 10.

[27] S. Aryal, A. S. Porawagama, M. G. S. Hasith, S. C. Thoradeniya, N. Kodagoda, and K. Suriyawansa, "MoocRec: Learning Styles-Oriented MOOC Recommender and Search Engine," *2019 IEEE Global Engineering Education Conference (EDUCON)*, May 2019, doi: 10.1109/EDUCON.2019.8725079.

[28] S. Lee, R. Girish, and Y. U. Kim, "Courserush a MOOC search engine.," 2017.

[29] S. Jiang, T. F. Hagelien, M. Natvig, and J. Li, "Ontology-Based Semantic Search for Open Government Data," in *2019 IEEE 13th International Conference on Semantic Computing (ICSC)*, Newport Beach, CA, USA: IEEE, Jan. 2019, pp. 7–15. doi: 10.1109/ICOSC.2019.8665522.

[30] Z. Pan, "Optimization of Information Retrieval Algorithm for Digital Library Based on Semantic Search Engine," in *2020 International Conference on Computer Engineering and Application (ICCEA)*, Guangzhou, China: IEEE, Mar. 2020, pp. 364–367. doi: 10.1109/ICCEA50009.2020.00085.

[31] B. Fazzinga and T. Lukasiewicz, "Semantic search on the Web".

[32] J. Douze H. and J. Johnson, "Faiss: A library for efficient similarity search," Jun. 28, 2018. https://engineering.fb.com/2017/03/29/data-infrastructure/faiss-a-library-for-efficient-similarity-search/ (accessed Nov. 10, 2022).

[33] M. Aumüller, E. Bernhardsson, and A. Faithfull, "ANN-Benchmarks: A benchmarking tool for approximate nearest neighbor algorithms," *Information Systems*, vol. 87, p. 101374, Jan. 2020, doi: 10.1016/j.is.2019.02.006.

[34] C. Qin, C. Deng, J. Huang, K. Shu, and M. Bai, "An Efficient Faiss-Based Search Method for Mass Spectral Library Searching,"

in *2020 3rd International Conference on Advanced Electronic Materials, Computers and Software Engineering (AEMCSE)*, Shenzhen, China: IEEE, Apr. 2020, pp. 513–518. doi: 10.1109/AEMCSE50948.2020.00116.

[35] A. Gupta, D. Agarwal, Veenu, and M. P. S. Bhatia, "Performance Analysis of Content Based Image Retrieval Systems," in *2018 International Conference on Computing, Power and Communication Technologies (GUCON)*, Greater Noida, Uttar Pradesh, India: IEEE, Sep. 2018, pp. 899–902. doi: 10.1109/GUCON.2018.8675107.

[36] J. Devlin, M.-W. Chang, K. Lee, and K. Toutanova, "BERT: Pre-training of Deep Bidirectional Transformers for Language Understanding." arXiv, May 24, 2019. Accessed: Feb. 14, 2023. [Online]. Available: http://arxiv.org/abs/1810.04805

[37] N. Reimers and I. Gurevych, "Sentence-BERT: Sentence Embeddings using Siamese BERT-Networks." arXiv, Aug. 27, 2019. Accessed: Feb. 15, 2023. [Online]. Available: http://arxiv.org/abs/1908.10084

[38] V. Sanh, L. Debut, J. Chaumond, and T. Wolf, "DistilBERT, a distilled version of BERT: smaller, faster, cheaper and lighter." arXiv, Feb. 29, 2020. Accessed: Mar. 04, 2023. [Online]. Available: http://arxiv.org/abs/1910.01108

[39] P. Bajaj *et al.*, "MS MARCO: A Human Generated MAchine Reading COmprehension Dataset." arXiv, Oct. 31, 2018. Accessed: Mar. 20, 2023. [Online]. Available: http://arxiv.org/abs/1611.09268

[40] N. Thakur, N. Reimers, A. Rücklé, A. Srivastava, and I. Gurevych, "BEIR: A Heterogenous Benchmark for Zero-shot Evaluation of Information Retrieval Models." arXiv, Oct. 20, 2021. Accessed: Feb. 14, 2023. [Online]. Available: http://arxiv.org/abs/2104.08663

[41] K. Jirvelin and J. Kekiiliinen, "IR evaluation methods for retrieving highly relevant documents," *SIGIR '00: Proceedings of the 23rd annual international ACM SIGIR conference on Research and development in information retrieval*, pp. 41–48, Jul. 2000, doi: https://doi.org/10.1145/345508.345545.

محرك البحث الدلالي الموحد متعدد اللغات MOOCs سريع وفعال (UMSSE)

أحمد فجر تاتانغ ، عبدالله القرني

قسم علوم الحاسبات، كلية الحاسبات وتقنية المعلومات، جامعة الملك عبد العزيز، جدة، المملكة العربية السعودية
atatang@stu.kau.edu.sa, amsalgarni@kau.edu.sa

المستخلص. أدى انتشار الدورات التدريبية المفتوحة عبر الإنترنت (MOOCs) إلى زيادة الطلب على محركات البحث الفعالة لمساعدة المتعلمين على تحديد الدورات التدريبية التي تلبي احتياجاتهم والتسجيل فيها. ومع ذلك، فإن بناء محرك بحث MOOC موحد متعدد اللغات يمكنه توفير نتائج بحث شاملة كان أمرًا صعبًا بسبب العديد من المنصات بلغات مختلفة وتنوع الدورات المتاحة. اقترحت هذه الورقة ونفذت نموذجًا لمحرك البحث الدلالي الموحد UMSSE(MOOCs). استفاد UMSSE من مجموعة من نماذج تشفير النص وخوارزمية تقريب الجار الأقرب (ANN) لتحسين سرعة ودقة نتائج البحث. قام النموذج بدمج البيانات من النظام الأساسي ولغات متعددة، باستخدام تقنيات معالجة اللغة الطبيعية (NLP) وتقنيات التعلم الآلي لفهم معنى استعلامات البحث والتوصية بالدورات ذات الصلة. تم تقييم أداء النموذج المقترح باستخدام مجموعة بيانات من أوصاف دورات MOOC المختلفة. أظهرت النتائج أنها تفوقت على محركات البحث التقليدية القائمة على الكلمات الرئيسية فيما يتعلق بمقاييس الأداء. وقد أوضحت العديد من الأمثلة التطبيقية كيف يمكن للنموذج المقترح أن يحسن سرعة وفعالية بحث MOOCs والتوصية بالدورات المناسبة للمستخدمين.

الكلمات المفتاحية. محرك البحث الدلالي، البيانات الوصفية، معالجة اللغة الطبيعية، الدورات التدريبية المفتوحة الموحدة عبر الإنترنت، تقنيات استرجاع المعلومات