

**IMPLEMENTATION OF AN INTELLIGENT  
INFORMATION SYSTEM RETRIEVAL SYSTEM**

**BY**

**OBOZOKHAE, EDEGHOGHON JOY  
(20CD028256)**

**A PROJECT SUBMITTED TO THE DEPARTMENT OF  
COMPUTER AND INFORMATION SCIENCES, COLLEGE  
OF SCIENCE AND TECHNOLOGY, COVENANT  
UNIVERSITY OTA, OGUN STATE.**

**IN PARTIAL FULFILMENT OF THE REQUIREMENTS  
FOR THE AWARD OF THE BACHELOR OF SCIENCE  
(HONOURS) DEGREE IN COMPUTER SCIENCE.**

**JULY, 2024**

## **CERTIFICATION**

I hereby certify that this project was carried out by EDEGHOGHON JOY OBOZOKHAE in the Department of Computer and Information Sciences, College of Science and Technology, Covenant University, Ogun State, Nigeria, under my supervision.

**Dr. Adubi Stephen**  
*Supervisor*

---

**Signature and Date**

**Prof. Olufunke O. Oladipupo**  
*Head of Department*

---

**Signature and Date**

## **DEDICATION**

I dedicate this work to God, my present help, and my parents, Mr. and Mrs. Obozokhae, for their constant care and support throughout my journey in this institution. I also dedicate this work to my friends and family, who support me.

## **ACKNOWLEDGEMENTS**

My most profound gratitude goes to Almighty God, who has kept and sustained me from the very start of my degree until now. To Him be all the glory. I want to thank my parents, who have always supported me sincerely. May their labour of love never be in vain I also want to sincerely thank my wonderful siblings, Owen Obozokhae, Ejemen Obozokhae, and Osemu Obozokhae, for always being there for me. May God bless you all

## TABLE OF CONTENTS

CONTENT	PAGES
COVER PAGE	i
CERTIFICATION	ii
DEDICATION	iii
ACKNOWLEDGEMENTS	iv
TABLE OF CONTENTS	v
LIST OF FIGURES	viii
LIST OF TABLES	ix
ABBREVIATIONS	x
ABSTRACT	xi
CHAPTER ONE: INTRODUCTION	1
1.1 Background Information	1
1.2 Statement of the Problem	2
1.3 Aim and Objectives of the Study	3
1.4 Methodology	3
1.5 Significance of the Study	5
1.6 Limitation of the Study	6
1.7 Project Organisation	6
CHAPTER TWO: LITERATURE REVIEW	7
2.1 Preamble	7
2.2 Review of Information Retrieval	7
2.2.1 History of Information Retrieval	7
2.2.2 Models of Information Retrieval	8
2.3 Architecture of an IR System	9
2.3.1 Indexing	9
2.3.2 Query Processing	9
2.3.3 Searching	10
2.3.4 Ranking	10
2.4 Review of Topic Modelling	10
2.5 Review of Latent Dirichlet Allocation	12
2.5.1 E-commerce Searching	12
2.5.2 Predictive Research	13
2.6 Review of relevant concepts	13
2.6.1 Artificial Intelligence	13
2.6.2 Machine Learning	13

2.6.3	Natural Language Processing	14
2.7	Review related methods	14
2.7.1	Non-Negative Factorization (NMF)	14
2.7.2	Biterm Topic Modelling (BTM)	15
2.7.3	Latent Semantic Indexing (LSI)	15
2.8	Review of Existing Systems	16
2.8.1	PubMed	16
2.8.2	Reddit	17
2.8.3	Google news	17
2.8.4	Zite	18
2.9	Summary of literature review	18
<b>CHAPTER THREE: SYSTEM ANALYSIS AND DESIGN</b>		<b>19</b>
3.1	Preamble	19
3.2	The Proposed System	19
3.3	Requirement analysis	19
3.3.1	Functional Requirements	19
3.3.2	Non-Functional Requirements	20
3.4	Data Collection	20
3.5	System Architecture	20
3.6	System Design	21
3.6.1	Physical Design	21
3.6.2	Logical Design	22
3.7	Algorithm Design	26
3.8	Description of tables	28
3.8.1	Document	28
3.8.2	DocumentTopic	28
3.8.3	Topic	28
3.8.4	TopicWord	29
<b>CHAPTER FOUR: SYSTEM IMPLEMENTATION AND EVALUATION</b>		<b>30</b>
4.1	Preamble	30
4.2	System Requirements	30
4.2.1	Hardware Requirements	30
4.2.2	Software Requirements	30
4.3	Implementation tools	31
4.3.1	Python	31
4.3.2	NLTK	32
4.3.3	Visual Studio Code	32
4.3.4	MySQL	32

4.4	Development Methodology	32
4.5	System Interfaces	32
4.5.1	The Home Page	32
4.5.2	The Upload Module	33
4.5.3	The Search Page	34
4.6	System Evaluation	34
4.6.1	Discussion	35
4.7	Usability testing	35
<b>CHAPTER FIVE: CONCLUSION AND RECOMMENDATIONS</b>		<b>38</b>
5.1	Summary	38
5.2	Recommendations	38
5.3	conclusions	38
<b>REFERENCES</b>		<b>39</b>

## LIST OF FIGURES

FIGURES	TITLE OF FIGURES	PAGES
2.1	Architecture of IR system	10
2.2	Pictorial description of NMF	15
2.3	Pubmed Interface	16
2.4	Reddit Interface	17
2.5	Google News Interface	18
3.1	Proposed system architecture	21
3.2	Use case diagram	23
3.3	Activity diagram of the proposed system	24
3.4	Sequence diagram of the proposed system	25
3.5	Entity relationship diagram of the proposed system	26
4.1	Home page of system	33
4.2	Upload module	33
4.3	The search page	34
4.4	Coherence and perplexityscore	34
4.5	Evaluation results	35
4.6	User question 1	36
4.7	User question 2	36
4.8	User question 3	36

## **LIST OF TABLES**

<b>TABLES</b>	<b>TITLE OF TABLES</b>	<b>PAGES</b>
1.1	Objective to Methodology Mapping	5
3.1	Documents table	28
3.2	Documents table	28
3.3	Documents table	28
3.4	Documents table	29
4.1	Hardware requirements table	30
4.2	Software requirements table	31

## **ABBREVIATIONS**

AI: Artificial Intelligence

IR: Information Retrieval

IRS: Information Retrieval System

LDA: Lantent Dirichlet Allocation

NLP: Natural Language Processing

VSM: Vector Space

## ABSTRACT

This paper introduces a web application that utilizes Latent Dirichlet Allocation for topic modelling to effectively manage and retrieve reports in an academic setting. The system allows uploaded reports to be analyzed using the LDA algorithm to derive topics, which can then be stored in a database for easy search, retrieval, and utilization.

The application was built with Flask for the backend, MySQL for database management, and a frontend developed with HTML, CSS, and Bootstrap. Additionally, NLTK was employed for natural language processing and Gensim for topic modelling. The performance of the LDA algorithm was evaluated using coherence and perplexity scores with user feedback.

Overall, this system enhances the efficiency of document retrieval and management, particularly in accessing past student projects for improved information retrieval processes within the department.

# CHAPTER ONE

## INTRODUCTION

### 1.1 Background Information

People have understood the value of information retrieval and archiving for thousands of years. Large volumes of information could now be stored thanks to the development of computers, and extracting relevant information from these repositories became essential (Roshdi & Roohparvar, 2015) . The two most significant developments in Information Retrieval have been the introduction of internet-based digital information and the widespread use of web search engines as retrieval tools since the early 1990s (Foster & Rafferty, 2011) . Information retrieval (IR) is the systematic process of collecting, storing, organising, and providing access to information resources. The ultimate goal is to facilitate seamless access for end-users, ensuring that the representation and organisation of information cater to their specific needs and requirements (Lal *et al.*, 2016).

Though widely used in various areas like search engines, databases, and digital libraries, traditional IR systems have shortcomings in understanding documents' semantic significance and context, which can result in incomplete or unrelated search outcomes. The inherent ambiguity of natural language, like English, challenges achieving exact matches between user queries and documents. This complexity complicates the retrieval process in the Dataspace (Lal *et al.*, 2016). Researchers have sought to overcome these limitations by exploring novel techniques and methodologies, leveraging the availability of extensive labelled datasets and advancements in computing power (Hambarde & Proenca, 2023)

One such technology that can improve search functionality is Natural Language Processing (NLP). NLP has been recognized as a powerful technique for advancing text interpretation and analysis by uncovering important semantic information in documents that cannot be identified through mere word counting. Recent NLP research has seen significant progress in information extraction, summarization, text-based IR, and multilingual translation (Zhou & Zhang, 2003). In addition, topic modelling, a subset of NLP, has become an essential method for uncovering concealed themes and subjects within extensive document collections, facilitating more precise and effective information retrieval.

Topic modelling involves using unsupervised Machine Learning to identify clusters or groups of similar words within a text, making it a type of statistical modelling . It involves identifying the words associated with the topics found within a document or a dataset. This

is valuable as extracting words directly from a document is more time-consuming and complex than extracting them from the topics within the document (Peddireddi, 2021). Latent Dirichlet Allocation (LDA) is one of the most commonly utilised algorithms in topic modelling.

Zhao *et al.* (2020) sees LDA as the building block in numerous machine-learning applications. LDA aims to categorize words and documents into specific clusters, also known as topics. These topics can then be utilised to categorize and retrieve documents efficiently (Maklin, 2022). With the use of LDA, every document can be expressed as a probabilistic distribution across latent themes, and every document's topic distribution has a common Dirichlet prior. To be more precise, it lets you assess how each word in your writing contributes to a specific topic and how the themes are distributed statistically. Each latent topic in the LDA model is similarly represented as a likelihood distribution over words, and the word distributions of the topics share an identical Dirichlet prior (Jelodar *et al.*, 2019). The algorithm estimates two sets of parameters: word distribution for each topic based on documents in the collection and topic distribution within the collection of documents (Kochmar, 2022). LDA has been effectively utilised in various fields, such as document categorisation, grouping, and retrieving information.

IR systems can significantly improve their search capabilities by including topic modelling. Using topic modelling, the system can comprehend the semantic linkages between words and documents beyond simple keyword matching. As a result, search efficiency and satisfaction for users are increased by more contextually relevant and accurate search results. Therefore, this study aims to develop an IR system integrating topic modelling to enhance search capabilities.

## **1.2 Statement of the Problem**

Organisations and individuals receive vast amounts of data in today's digital age. Most traditional IR approaches have proven ineffective and inappropriate for organisations and individuals because they rely on keyword-based search algorithms. In this regard, finding the underlying themes and contexts in documents is beyond the capability of these systems, making data retrieval and management inefficient. Additionally, current IR systems do not usually allow users to upload their documents for personalised analysis, offering minimal opportunity for users to manage search results according to their organisational needs. This poses problems for users categorising, organising, and retrieving documents relevant to their unique contexts.

This study presents the development of an IR system with a critical advanced feature: support for topic modelling techniques, such as LDA, applied to understanding and extracting thematic structures from large text corpora. The system will allow users to upload documents, apply topic modelling, and store those results in a database for later searching. This can yield more relevant, context-aware search results and enhance the efficiency and effectiveness of general information retrieval processes within organisations. The current project aims to do this by developing an intelligent information retrieval system that incorporates topic modelling in analysing documents uploaded by the user. This study seeks to improve the retrieval and organisation of documents, making it easier for users to locate and manage information relevant to their specific needs and contexts.

### **1.3 Aim and Objectives of the Study**

This project aims to implement an intelligent information retrieval system to manage departmental reports. The objectives of the study are:

- (i) To gather and identify requirements for the system.
- (ii) To model and design the components of the intelligent information retrieval system.
- (iii) To implement a working intelligent information retrieval system prototype.
- (iv) To evaluate the performance of the system.

### **1.4 Methodology**

The research methodology further explains the ideal approach to achieving the objectives.

#### **(i) Objective 1: To gather and identify requirements for the system.**

It is widely acknowledged that requirements form the basis of any system; hence, an extensive literature review will be carried out to understand the needs and expectations of intelligent information systems successfully. Additionally, surveys will be conducted to gather information from potential users regarding their preferences, needs, and expectations.

#### **(ii) Objective 2: To model and design the components of the intelligent information retrieval system.**

This will involve using various UML diagrams, such as the use case diagram, activity diagrams, and sequence diagram, to model and develop the system's components.

**(iii) Objective 3: Implement a working intelligent information retrieval system prototype.**

Python Flask will be used for the system's backend to successfully implement the system because of its lightweight and flexible architecture, making it ideal for building scalable and efficient web applications. The database for the system will be created using MySQL because of its reliability and performance; it is also highly scalable and can handle large amounts of data. Genism, a popular Python library, will be used for topic modelling because of its ease of use, scalability, and extensive documentation. Natural Language Toolkit (NLTK), another Python library, will be used for data preprocessing and cleaning; it is easily integrated with other libraries, such as Genism, making it a suitable choice. HTML, CSS, and Bootstrap will be utilised for the system's front end.

**(iv) Objective 4: To evaluate the performance of the system.** The system will be evaluated based on coherence, perplexity scores, and user feedback.

**Table 1.1: Objective to Methodology Mapping**

<b>S/N</b>	<b>Objectives</b>	<b>Methodology</b>
1	To gather and identify requirements for the system.	<b>Literature review and survey</b> conduct an extensive literature review and carry out surveys on concerned stakeholders.
2	To model and design the components of the intelligent information retrieval system.	<b>System design and modelling</b> Modelling of requirements using UML diagrams: Use-case, Activity, and Sequence Diagrams.  Database Design Modelling Using Entity Relationship Diagrams.  System Architecture design using lucid chart.
3	To model and design the components of the intelligent information retrieval system. To implement a working prototype of the system.	<b>Implementation and prototype development</b> Interface design using HTML, CSS, and Bootstrap  Topic modelling using Gensim  Backend development using Python Flask
4	To evaluate the performance of the system.	<b>Evaluating system performance</b>  Evaluate the system based on coherence and perplexity scores and user feedback.

### 1.5 Significance of the Study

The significance of this web application for intelligent information retrieval is relatively high, especially in a place like an academic community. The significant advantage of this project is that it will give better management and efficient access to past research papers accomplished by students. An advanced topic modelling algorithm will categorise and make such research documents searchable based on their content. It is a system that addresses an issue common to students and faculty: the big challenge in searching for relevant past research work. The traditional methods are redundant and, most of the time, not practical since it always takes much time to find a document from extensive archives of research papers. With this new system, users can upload their documents; these documents will go through processing, whereby the key topics will be identified easily. This improves docu-

ment organisation and drastically reduces the time spent looking for research, facilitating an effective and more productive research process.

## **1.6 Limitation of the Study**

The effectiveness of the topic modelling approach is often constrained by the size and diversity of the dataset used to train the model. When the dataset is relatively small, it may not provide enough variability and richness for the model to uncover meaningful and insightful topics.

## **1.7 Project Organisation**

Chapter One of the project contains an explanation of the project, problems with existing solutions, the need for an improved solution, the method of implementation, the significance of the study, and the limitations. Chapter Two describes the existing systems related to the project topic and the methodology, algorithm, and techniques used in related systems. Chapter Three describes the analysis and system design. Chapter Four shows the implementation of the system in detail and the results obtained. Chapter Five summarises the project and gives recommendations, suggestions, conclusions, and references

## **CHAPTER TWO**

### **LITERATURE REVIEW**

#### **2.1 Preamble**

The research investigates LDA, a widely used algorithm for topic modelling in NLP, can be used as the basis for creating an intelligent document retrieval system. Using the latent semantic structure present in textual data, LDA makes it possible to automatically identify underlying subjects in project papers, which makes information organisation, classification, and retrieval more efficient.

#### **2.2 Review of Information Retrieval**

The following section contains vital concepts on IR.

##### **2.2.1 History of Information Retrieval**

The concept of information retrieval did not just begin with the internet; before the high rise in search engines, IR systems were used in commercial and intelligence applications as far back as the 1960s. Antecedent to that, the first computer-based search system was built in the late 1940s (Sanderson & Croft, 2012). A Univac machine could store specific signals on magnetic steel tape, retrieve a document associated with those symbols, and rewrite its contents. In the 1970s, information retrieval systems could effectively handle several thousand documents. However, significant changes occurred after 1992, following the Text Retrieval Conference, supported by the US Department of Defense and the National Institute of Standards and Technology.

In the early 1990s, computer power and storage experienced a significant rise accompanied by lower prices. A notable achievement in the early 1990s was the release of Archie by Peter Deutsch, Alan Emtage, and Bill Heelan at McGill University. Archie was a “search engine” that let users access a particular website (an Archie server) and use command lines to look for files from open file-sharing websites that were previously gathered for that server (or other Archie servers that were connected to it) (Harman, 2019). The years following saw the release of other several notable systems including the WAIS (Wide Area Information Server) by Brewster Kahle and Harry Morris, WAIS was a client-server full-text searching system that indexed and searched datasets on WAIS servers using the ANSI Standard Z39.50. In that same decade, Gopher was released by Paul Lindner and Mark P. McCahill from the University of Minnesota; Gopher was a protocol that allowed documents to be

stored, indexed using menus or directories, and connected to other Gopher servers (Harman, 2019).

### **2.2.2 Models of Information Retrieval**

Hiemstra (2009), in a research paper, says that IR models are helpful for two reasons. First, models serve as a guide for research and a forum for scholarly discourse. The ability to use models as a template for actual retrieval system implementation is the second reason for using models. The section below highlights relevant models for IR.

#### **A. Boolean Model**

The Boolean model, one of the earliest and most straightforward retrieval techniques in information retrieval systems, operates on the principles of set theory and Boolean algebra. In this model, documents are characterised by linked index terms, and the retrieval process involves matching the terms in a user query with the index terms assigned to the documents (Lashkari *et al.*, 2009). The Boolean Retrieval Model is a fundamental information retrieval model that is generic but can be adapted to various formats. It relies on Boolean logic and uses term-document matrix or inverted index data structures (Ramishamukhtar, 2023). Giving (expert) users a sense of control over the system is one benefit of the Boolean model. Given a query, it is immediately evident why a document has been retrieved. If the final document set is too small or large, it is obvious which operators will produce a larger or smaller set. Several evident drawbacks of the model exist for inexperienced users. Its primary drawback is that it does not offer a ranking of the documents that are retrieved. The model may result in the system making somewhat irritating judgments because it can either obtain a document or not (Hiemstra, 2009).

#### **B. Vector Space Model (VSM)**

In 1975, G. Salton introduced a model that substantially impacted the information retrieval field. The main objective of his approach is to fetch text documents automatically. This model depicts queries and documents as vectors containing weighted words (Larson, 2011). The vector space model is a mathematical model used to represent text documents and various types of multimedia objects as vectors consisting of identifiers like index terms. This model relies on the similarity between the search document and the query created by the user, which should closely match the documents required for information retrieval (Sudheer, 2022). The idea behind vector space modelling is that it is possible to compute the similarities between queries and terms or documents by arranging terms, documents, and queries in a term-document space. This allows the computation results to be ranked based

on the similarity measure between them. The VSM makes it possible to determine comparable documents (Singh, 2022).

### **C. Probabilistic Model**

Roberston and Sparck Jones presented the traditional probabilistic model, which later evolved into the binary independence retrieval (BIR) model, which is an information retrieval approach that represents documents and queries as binary vectors, where each element indicates the presence or absence of a term, in 1976. It is based on the Probability Ranking Principle, which states that “an information retrieval system is supposed to rank the documents based on their probability of relevance to the query, given all the evidence available,” the probabilistic model aims to capture the IR problem within a probabilistic framework (Shade *et al.*, 2012). Enhancing the probabilistic description of the optimal answer set of documents associated with a query through a sequence of interactions is the fundamental idea behind a probabilistic model (Ribeiro & Muntz, 1996). The following is the procedure for a query: Initially, the user looks at the recovered documents and estimates which ones are connected and which are not. This provides a rough probabilistic description of the optimal set of answers. Using this information, the system can improve the optimal answer set’s description. As this procedure is repeated numerous times, it is anticipated that the description will change and get closer to the actual description of the optimal response set (Dong *et al.*, 2008).

## **2.3 Architecture of an IR System**

An IR system comprises several modules that help users find the information they need for extensive data collection. They include the following:

### **2.3.1 Indexing**

This is a crucial step in the IR process where documents are represented in their original state content form. It creates a term index that links to documents that are stored. It parses text from documents to locate terms, storing text in units known as terms for precise and quick retrieval (Larson, 2011). It simply involves classifying and arranging resources or information that is readily accessible and searchable. Users can find pertinent content more quickly and effectively with indexing.

### **2.3.2 Query Processing**

The process of converting an unprocessed query into a processed query using several techniques, such as expansion and refinement, is called query processing. A processed query

needs to explain the user's information needs fully. The system handles user queries, including keywords, phrases, or advanced search expressions, and compares them with indexed documents to find appropriate results (Islam, 2024)

### 2.3.3 Searching

The search process is one of the main components of IR architecture. The IR system gets documents containing the query keywords during the search process. The retrieval process involves comparing the query phrases in the information resources with the documents. Index files direct users to related documents throughout the search process. The particular outcomes of a search are contingent upon the chosen infrared model.

### 2.3.4 Ranking

The ranking of retrieved documents in IR is considered an issue of essential concern because it directly impacts the effectiveness and efficiency of the retrieval system (Sharma *et al.*, 2022). After potential matches are found, the system sorts them according to their relevance to the user's search. Different algorithms and criteria prioritise search results, including keyword frequency, document popularity, and relevance feedback (Islam, 2024).

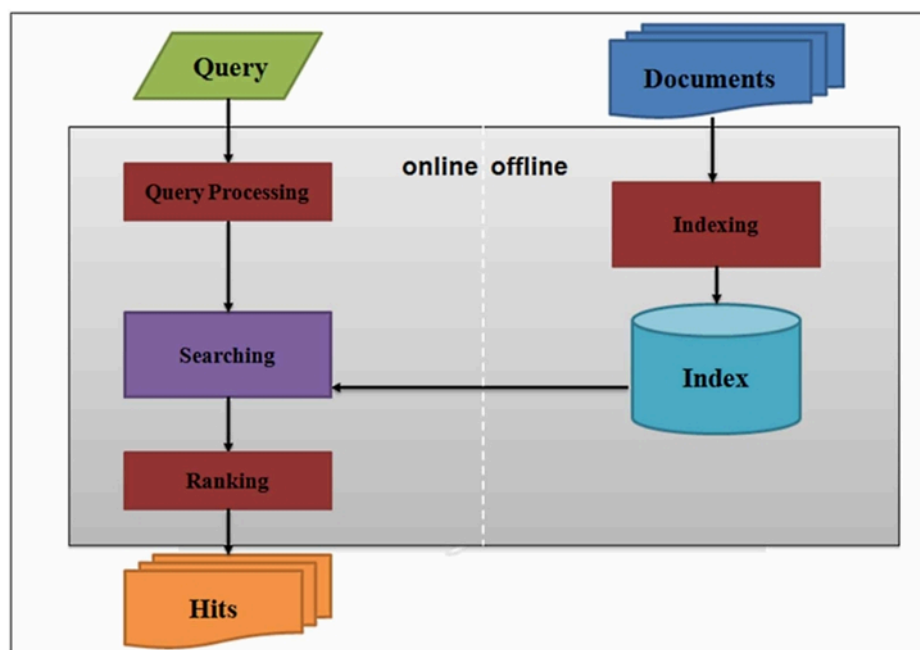


Figure 2.1: Architecture of IR system

## 2.4 Review of Topic Modelling

Over the past few years, there has been a noticeable surge in the accumulation of digital data daily. However, not all this data holds immediate utility; the imperative lies in uncovering latent insights or valuable information concealed within this vast volume of data. Extracting

meaningful relationships and pertinent information from such extensive datasets presents a formidable challenge . Topic modelling is a highly efficient text-mining technique for extracting latent information and revealing correlations between textual documents and data in data mining.

Topic modeling is an unsupervised machine learning technique that can analyze a set of documents, identify patterns in the words and phrases used, and automatically group the documents into related topics. This method allows for the discovery of underlying themes and concepts within a large corpus of text, without the need for predefined categories or labels (Pascual, 2019). It is also seen as a probabilistic method that can be used to measure the hidden structures in a document is. It Considered a method within natural language processing and machine learning, it strives to recognise subjects or themes in a set of documents autonomously. It is an unsupervised learning meaning it does not rely on labelled training data but instead identifies patterns and structures within the text data.

In recent years, topic modelling has surged in popularity, permeating diverse sectors of life and becoming a ubiquitous tool across various domains. Social media platforms have become essential data providers. A lot of topic modelling research has concentrated on using these data. A research article by Cheng *et al.* (2014) shows that Topic modelling has a profound influence on social media platforms as it allows for the analysis of extensive short texts, including tweets and microblogs. Analysts can reveal underlying topics in social media posts through methods like BTM (Biterm Topic Model), thus enabling analysis. Another article by Alkhodair *et al.* (2018) argues that Topic modelling is comprehensive and essential for social media platforms as it helps to understand the content better, improve topic representation using methods such as Twitter-LDA, WordNet, and hashtags, and enhance predictive performance, particularly for short and noisy micro-posts. In general, topic modelling on social media platforms helps organise content, analyse trends, and develop user interaction strategies by identifying essential topics and improving the comprehension of written information.

In the field of medical and biomedical science, the use of topical modelling is not only essential but also very promising. With the rapid increase in data in this area, including clinical records, genomic sequences, scholarly publications, and electronic health records, topical modelling effectively discovers hidden patterns, connections, and valuable insights within this extensive and intricate information environment. Research by Coroiu & Nutu (2019) presents a method proposed to aid physicians in diagnosing patients by analysing

their medical records. It underscores the significance of medical records in providing pertinent feedback for future health concerns based on a patient's medical history. The paper introduces a technique based on topic modelling and document clustering to automate extracting relevant information from these records. The objective is to categorise interconnected works into clusters representing relevant topics.

Internet repositories have become the predominant medium for accessing research articles. The availability of pertinent search results is crucial when seeking specific research papers. Hence, employing topic modelling can prove advantageous in extracting valuable information. Moreover, extending topic modelling to recommend analogous articles and documents enhances its utility (Apte *et al.*, 2011).

## **2.5 Review of Latent Dirichlet Allocation**

The concept of LDA was first introduced by Blei *et al.* (2003). LDA emerged as a probabilistic generative model tailored for textual corpora or documents. LDA is a prevalent technique in topic modelling and is widely preferred by many researchers. This method recognises critical terms within the dataset, creating a Bag of Words (BOW) for extracting features. BOW generates a dictionary of words from these terms, enabling focused content analysis without going through the entire document. This method helps to establish links between the collected dataset and results, which are then displayed in statistical and visual formats (Sharma *et al.*, 2022). It portrays every text document as a blend of topics, with each topic symbolising a distribution of words. LDA is a robust algorithm widely used in many fields to uncover hidden meanings in written information. It breaks down documents into different topics, helping us understand the main ideas and use this knowledge in different areas. A few of these fields include:

### **2.5.1 E-commerce Searching**

In a research paper by Yu *et al.* (2014), they propose a novel approach for enhancing e-commerce search results on eBay by using LDA to diversify the retrieved items based on hidden user intents. The approach addresses clutter and redundancy issues in search results by ranking diverse items for relevance and information novelty. A new metric called average satisfaction is introduced to measure user satisfaction with the search results, and empirical evidence demonstrates higher user satisfaction with the LDA-based approach compared to other diversified retrieval methods and eBay's production ranker. The article discusses the difficulties in conducting product searches in e-commerce, such as unclear search queries and disorganised product categories. It introduces the LDA-based method as a potential

solution to these challenges. The LDA model effectively identifies relevant user intentions, increasing user satisfaction.

### **2.5.2 Predictive Research**

Another research article by Gupta *et al.* (2022) titled “Prediction of research trends using LDA-based topic modelling” shows that the text classification technique, which involves identifying the subject of a text based on its content, is widely utilised. The LDA model has been utilised to analyse research trends in various fields, such as Applied Intelligence, by determining the topic of a document. Additionally, it has been employed to forecast the future growth of different areas within Applied Intelligence using statistical models that capture the linear relationship between multiple entities over time.

## **2.6 Review of relevant concepts**

This section details the various concepts that are relevant to this project. They include:

### **2.6.1 Artificial Intelligence**

Artificial intelligence (AI) is the area of computer science focused on creating intelligent machines that can perform tasks typically requiring human intelligence. This encompasses capabilities like learning, reasoning, problem-solving, perception, and language understanding. AI entails crafting algorithms and models to empower machines to imitate cognitive functions, emulating human-like intelligence. AI systems typically ingest large amounts of labelled training data, which they then analyze to identify correlations and patterns within the data. These discovered patterns are then used by the AI system to make predictions about future states or outcomes.

### **2.6.2 Machine Learning**

Machine learning is a branch of AI that focuses on creating systems that learn from the data they utilise and enhance their performance. In AI, machine learning focuses on developing statistical models and techniques that let computer systems learn and grow without explicit programming. It is the study of how computers learn without being specifically programmed. Several methods are used in machine learning to address data-related issues. Data scientists emphasize that there is no single algorithm that is universally optimal for solving all problems. The choice of method depends on the specific problem at hand, the number of variables involved, the type of model that works best, and other relevant factors. The most suitable algorithm to address a given problem is not a one-size-fits-all solution (Mahesh, 2018). Several algorithms are applicable in machine learning, including linear

regression algorithms, multivariate regression analyses, logistic regression, support vector machines, etc.

### **2.6.3 Natural Language Processing**

NLP is a field within AI that focuses on enabling computers to comprehend, generate, and work with human language. NLP provides the capability to interact with and extract information from data using natural language input, whether in the form of text or voice. Researchers want to learn more about how people perceive and utilise language to create suitable tools and methods to enable computer systems to comprehend and manipulate natural languages to carry out the necessary activities. Numerous academic disciplines, including machine translation, natural language text processing and summarisation, user interfaces, multilingual and cross-linguistic information retrieval (CLIR), speech recognition, artificial intelligence and expert systems, and more, use NLP in applications (Chowdhury, 2003).

## **2.7 Review related methods**

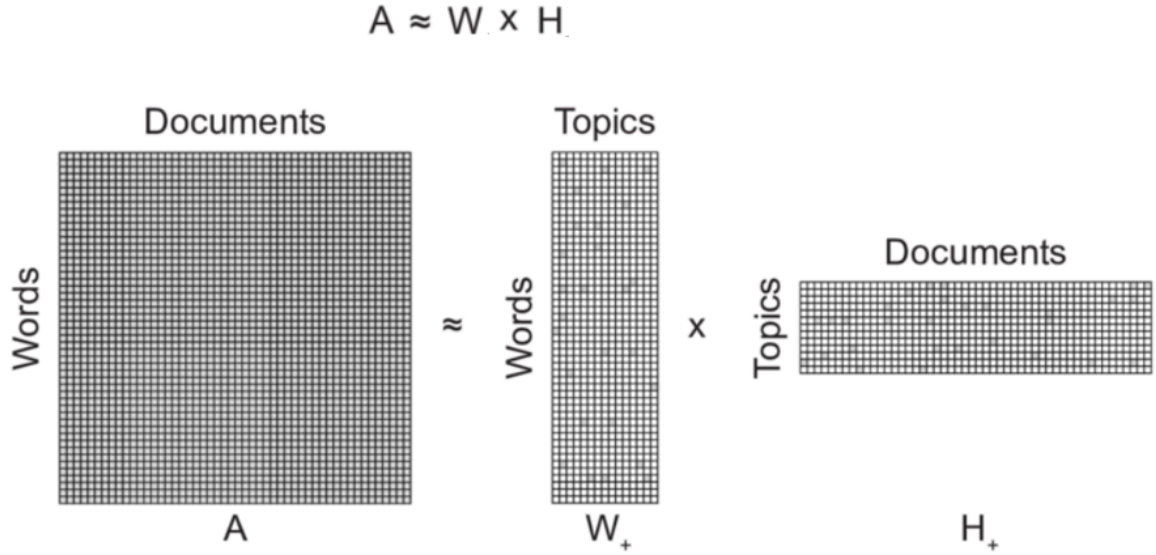
This session delves into alternative approaches within the realm of topic modelling, encompassing the following methodologies.

### **2.7.1 Non-Negative Factorization (NMF)**

Non-Negative Matrix Factorization is a statistical technique employed to diminish the complexity of input datasets or data sets, aiming to condense their dimensions. Using factor analysis, it assigns lower significance to words lacking coherence. It represents a novel approach to reducing dimensions, tackling the issue of negative numbers within the dataset by imposing constraints that ensure non-negativity on the data model (Kherwa & Bansal, 2020). Non-negative Matrix Factorization (NMF) is a widely adopted data representation and clustering method, extensively applied across various domains of machine learning and data analysis. NMF effectively condenses the features of each sample into a vector, approximating it through a linear combination of basis vectors to attain low-dimensional representations. However, in practical scenarios, features often possess varying degrees of importance. When it comes to NMF, some important points should be noted, which include the following:

- (i) It is a part of the group of linear algebra techniques utilised for discovering the underlying or concealed patterns within the dataset.
- (ii) It is represented as a non-negative matrix.

- (iii) It can also be used for Topic Modelling by inputting the term-document matrix, which is usually normalised using TF-IDF.
- (iv) NMF's popularity has grown due to its capability to derive factors that are sparse and easily understandable automatically.



**Figure 2.2: Pictorial description of NMF**

### 2.7.2 Biterm Topic Modelling (BTM)

Topic modelling over short texts is becoming increasingly significant because short writings are standard online. These days, short sentences are standard on the web, especially with the rise of social media. It becomes crucial to infer subjects from large-scale brief writings. Hence, BTM was introduced, a revolutionary topic model for brief writings (Jelodar *et al.*, 2017). In an article, Cheng *et al.* (2014) introduced the term topic model BTM, a revolutionary topic model for brief writing by explicitly modelling word co-occurrence patterns throughout the whole corpus; this approach may effectively capture subjects within brief texts. According to their findings, BTM produces less cohesive themes in short texts and discriminative topic representations. For several brief text analysis applications, including social media text mining, customer review analysis, and news item categorisation, BTM has been extensively utilised.

### 2.7.3 Latent Semantic Indexing (LSI)

The increasing volume of data collected has led to a growing need for efficient information retrieval and database search capabilities. Relying solely on lexical matching techniques can produce irrelevant or inaccurate results due to the presence of synonyms and words with multiple meanings. One approach to address this challenge is LSI, an information retrieval

method that uses a vector-space model to identify documents most relevant to a user's query, ranked by their similarity to the query (Witter & Berry, 1998). The process of examining a collection of documents to find statistical co-occurrences of words that occur together, which subsequently provides insights into the themes of those words and documents, is known as latent semantic indexing (Boyd, 2018).

## 2.8 Review of Existing Systems

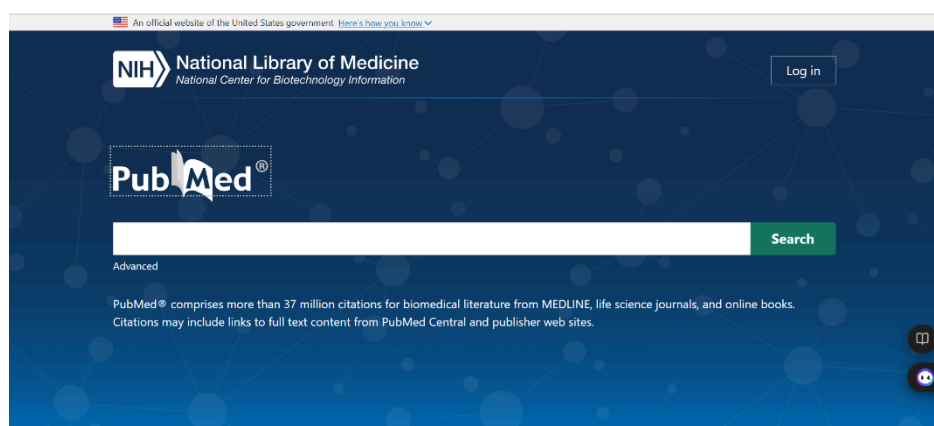
In recent years, various systems have been developed to incorporate LDA into their operations and other topic modelling algorithms for several functions, including text analysis, information retrieval, etc. These systems use topic modelling to discover hidden themes and patterns in large amounts of text, making information management more efficient and effective. Here is a brief overview of some notable systems and applications that use LDA and topic modelling in their operations:

### 2.8.1 PubMed

PubMed is a free search engine focusing on biomedical and life sciences literature. It uses topic modelling, specifically LDA, to improve its search and retrieval functions. This helps users find relevant research articles more effectively by categorising and analysing the content.

#### A. Key features

- (i) Enhanced search accuracy by implementing topic-based indexing.
- (ii) Clustering of similar research articles.
- (iii) Enhanced navigation of vast biomedical literature



**Figure 2.3: Pubmed Interface**

## 2.8.2 Reddit

Reddit, a widely used social media platform, uses LDA to analyse and categorise the content created by users in different subreddits. This helps to organise discussions and enhance the recommendations for content.

### A. Key features

- (i) Automated categorisation of posts and comments
- (ii) Enhanced content recommendation based on identified topics
- (iii) Enhanced user interaction by providing meaningful content recommendations.

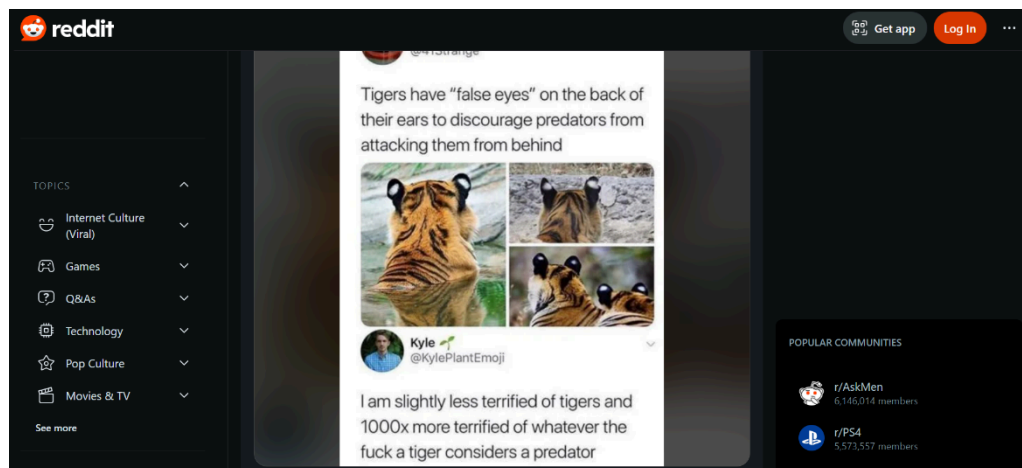


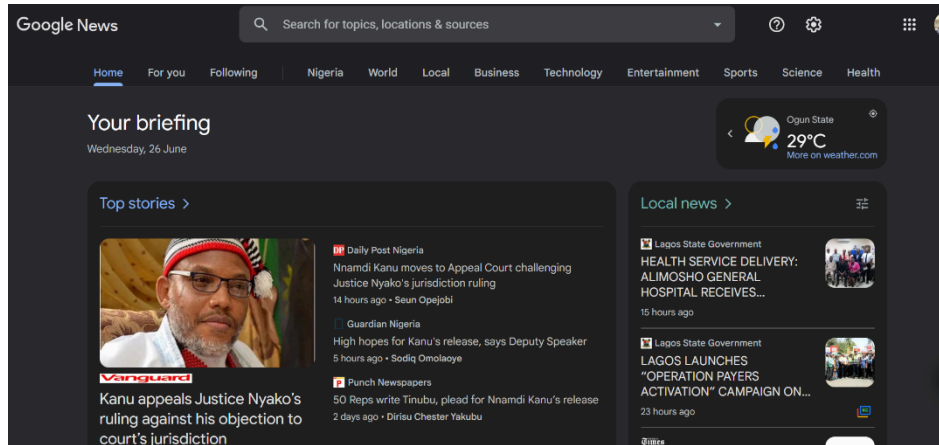
Figure 2.4: Reddit Interface

## 2.8.3 Google news

Google News uses LDA to gather news articles from different sources and organise them into specific topics. Using LDA, the system can recognise and group similar news stories. This helps users stay informed about the latest events and trends by providing a well-rounded perspective on the news.

### A. Key features

- (i) Automated categorisation of news articles.
- (ii) Collection of interconnected articles organised by shared themes.
- (iii) Improved user engagement by providing customised news updates tailored to individual wants.



**Figure 2.5: Google News Interface**

#### **2.8.4 Zite**

The system mentioned is a content recommendation system that uses LDA for topic modeling. It suggests articles to users depending on their interests and preferences, determined by their interaction history. LDA examines the content of articles, organises them into topics, and aligns them with user preferences to create customised suggestions. The system consistently revises user profiles and enhances recommendations using user input, enhancing accuracy as time passes.

### **2.9 Summary of literature review**

This literature review encompasses the development and the importance of information retrieval systems and topic modeling techniques. Advanced IR models, such as the Boolean model, VSM, and the probabilistic models, have significantly enhanced accuracy and relevance in searching. It reviews how LDA is highly effective in discovering latent patterns across large text datasets, which has yielded better document categorization and recommendations.

## **CHAPTER THREE**

### **SYSTEM ANALYSIS AND DESIGN**

#### **3.1 Preamble**

This section describes the suggested system's analysis and design in detail. It includes the system requirements, the methods used to implement it, and diagrams representing the suggested system.

#### **3.2 The Proposed System**

The proposed system will be a web-based application designed to facilitate the management and retrieval of report papers based on their content. The application will utilise a topic modelling algorithm, specifically LDA, to analyse the abstracts of uploaded research papers and store the derived topics in a database.

#### **3.3 Requirement analysis**

The Requirements Analysis process establishes users' anticipated needs and desires for a new or modified application. Requirements analysis encompasses the activities aimed at understanding the needs of various stakeholders. In essence, it involves analysing, documenting, validating, and managing the requirements of software or systems.

##### **3.3.1 Functional Requirements**

Functional requirements refer to the tasks the system must carry out to meet the needs of the stakeholders. They refer to the intended functions of a program or system, as specified in software development and systems engineering. For this proposed system, the functional requirements are as follows:

- (i) Document Upload: Users should be able to upload abstracts and research papers in different file formats (e.g., PDF, DOCX).
- (ii) Topic Modelling: The system shall apply the Latent Dirichlet Allocation (LDA) algorithm to the documents to identify and classify the main topics.
- (iii) Search Functionality: The user should be able to search for research papers by topic.
- (iv) Efficient Retrieval: The system will facilitate adequate storage and retrieval processes, guaranteeing fast access to stored data.

### **3.3.2 Non-Functional Requirements**

Nonfunctional requirements, or NFRs, consist of criteria defining the system's operational abilities and limitations. The non-functional aspects of this system are as follows:

- (i) Performance: Both uploads and searches should be completed in a reasonable amount of time.
- (ii) User Interface: The user interface should be easy to use and user-friendly, enabling users to effortlessly upload documents, conduct searches, and quickly understand search outcomes.
- (iii) Reliability: The users should have access to the application a high percentage of the time.
- (iv) Scalability: The system should be able to manage higher levels of traffic and uploads of documents without experiencing a noticeable decrease in performance.

### **3.4 Data Collection**

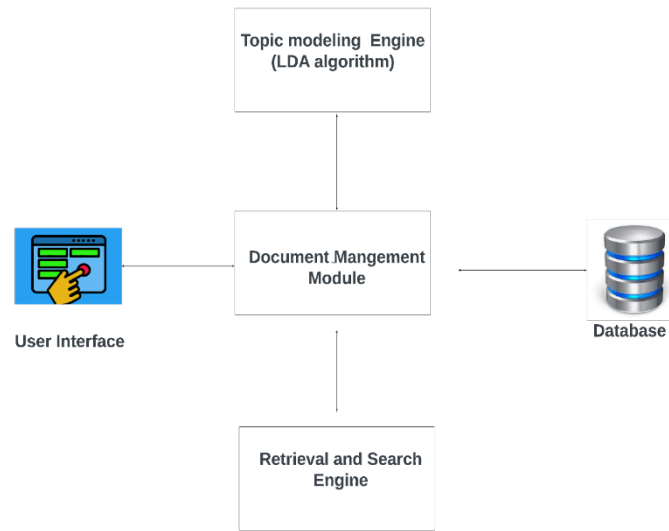
The collected data for the development of the LDA web application by creating a Google Form for students to submit their project reports. The form was designed to gather detailed information such as the report title, abstract, and full text. This data was used as the main input for the topic modeling process, allowing students to search and access documents based on the generated topic models. Using this method made it easy for students to participate and provided a rich dataset for thorough topic analysis.

### **3.5 System Architecture**

System architecture encompasses a system's complete arrangement and structure, comprising its elements, interconnections, and the fundamental rules and standards that dictate its creation and development. A system architecture is a mental model that describes a system's structure, behaviour, and other capabilities. The system's architecture will help with the logic and understanding of the system's design and model usage. The proposed system architecture for the intelligent document retrieval system involves several components, including the user interface, the database, the topic modelling engine, the document upload mechanism, and the search and retrieval engine.

- (i) User Interface: Users interact with the system through a web browser. The interface is designed to be user-friendly and responsive.
- (ii) Document Management module: Manages the upload of research papers of research papers, Ensuring the correctness of file formats and dimensions.

- (iii) Topic modelling engine: Extracts and processes the documents uploaded and applies the Latent Dirichlet Allocation (LDA) algorithm to discover the primary topics found within the document.
- (iv) Search and Retrieval Engine: Manage user queries, fetch appropriate database documents, and present concise and structured search outcomes.
- (v) Database: Stores documents path, topic modelling results, and user data.



**Figure 3.1: Proposed system architecture**

## 3.6 System Design

In this section, the focus will be on the two primary aspects of system design: physical design and logical design. The physical design focuses on the hardware and software infrastructure needed to implement the system. On the other hand, the logical design addresses the conceptual blueprint of the system, detailing how different components interact, the flow of data, and the overall architecture

### 3.6.1 Physical Design

Physical design focuses on the actual input and output processes of a system. It addresses how data is entered, validated, processed, and presented as output. The physical design stage produces the working system by defining the detailed design specifications that precisely describe the system's functionality. Key aspects of physical design include user interface design, process design, and data design.

### **A. Input Design**

Input design refers to how we create and maintain how data goes into a system. The goal of input design is to ensure that the data collected is accurate, complete, and easy to use. The effect of input design mainly affects data quality and user satisfaction; a sound input design is crucial to the success of any system. This system's input design will use a document upload interface where users can select research papers from their local storage to upload.

### **B. Output Design**

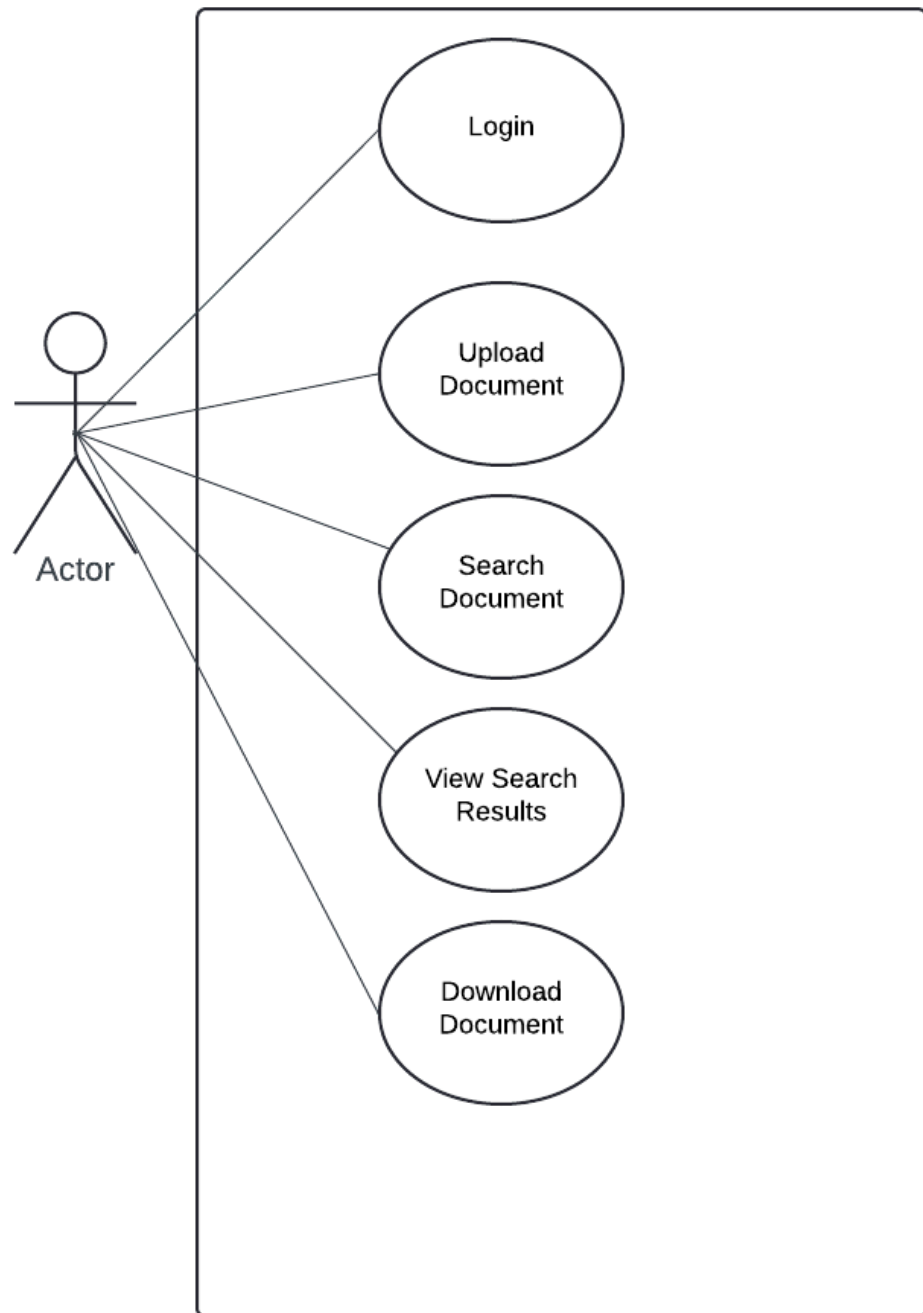
This determines how a computer system will communicate information to the user in a user-friendly and understandable form. An output design should be clear, concise, and functional. This is achieved by presenting the information in a form easily read and understood by the system users to improve user-system interaction.

## **3.6.2 Logical Design**

The logical design phase describes the operations and data structures required for the system without detailing how they will be implemented. This stage involves identifying key business entities and their relationships and creating a conceptual model that outlines the overall structure and behaviour of the system. As this logical design is refined and transformed into a physical design, specific technical details and implementation decisions are made, such as choosing specific databases, programming languages, and hardware platforms.

### **A. Use Case Diagram**

Use case diagrams (UCDs) are a popular tool for capturing the requirements and desired functionality of software products, providing a visual representation of the interactions between users and the system, and helping to identify the key features and behaviours that the software should exhibit.



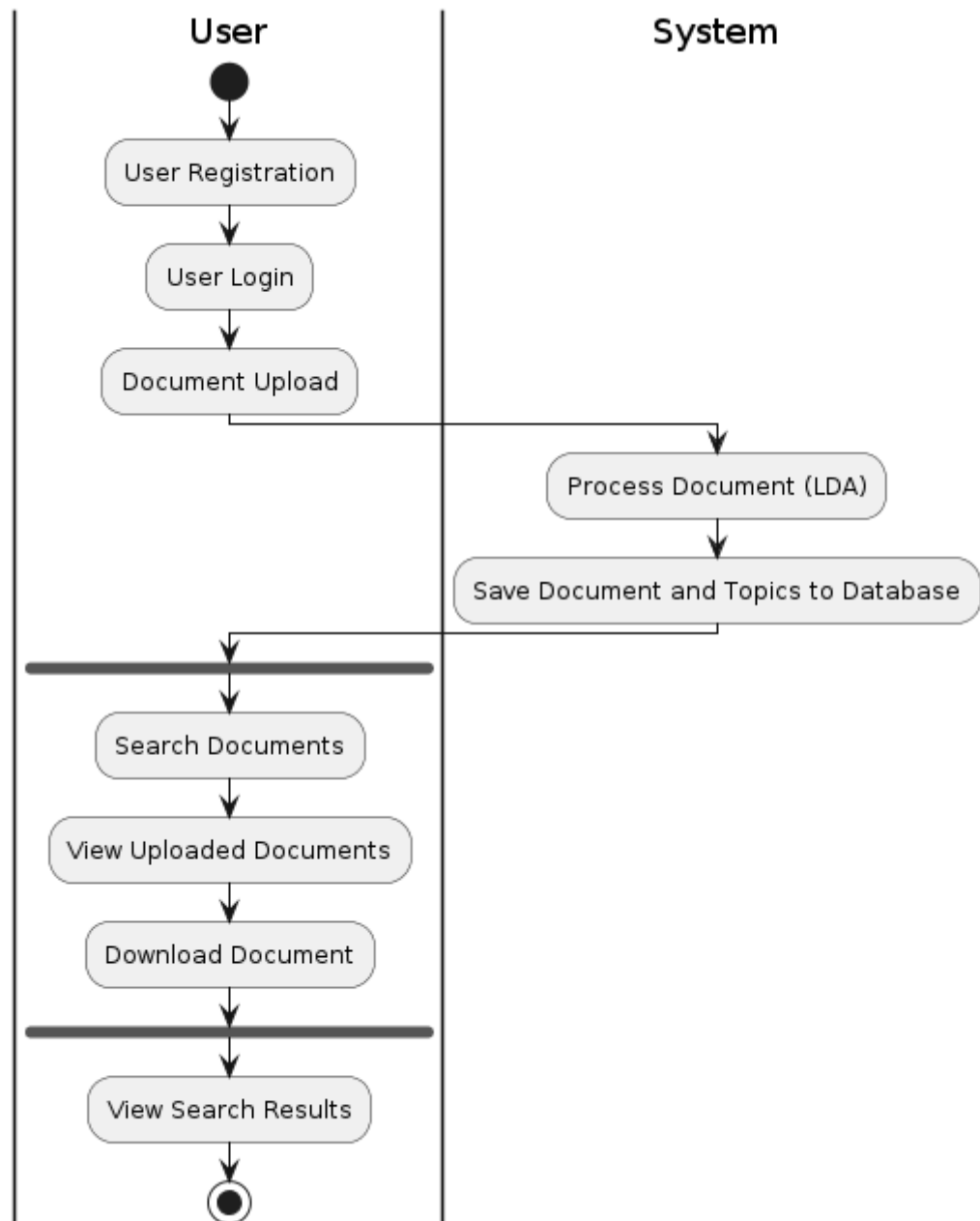
**Figure 3.2: Use case diagram**

The use case diagram illustrated in Figure 3.2 shows the different actions the user can take, which include the following:

- (i) Upload document: Users can upload research papers into the system.
- (ii) Search document: Users can search for topics based on relevant topics.
- (iii) View search results: Users can view relevant search results.
- (iv) Download document: Users can download relevant documents.

## B. Activity diagram

An activity diagram offers a valuable visualisation of a system's inner workings. It provides a clear and concise representation of the sequential flow of activities, enabling a deeper understanding of the underlying process and how each step contributes to the overall system's functionality. The figure below shows the activity diagram of the proposed system.

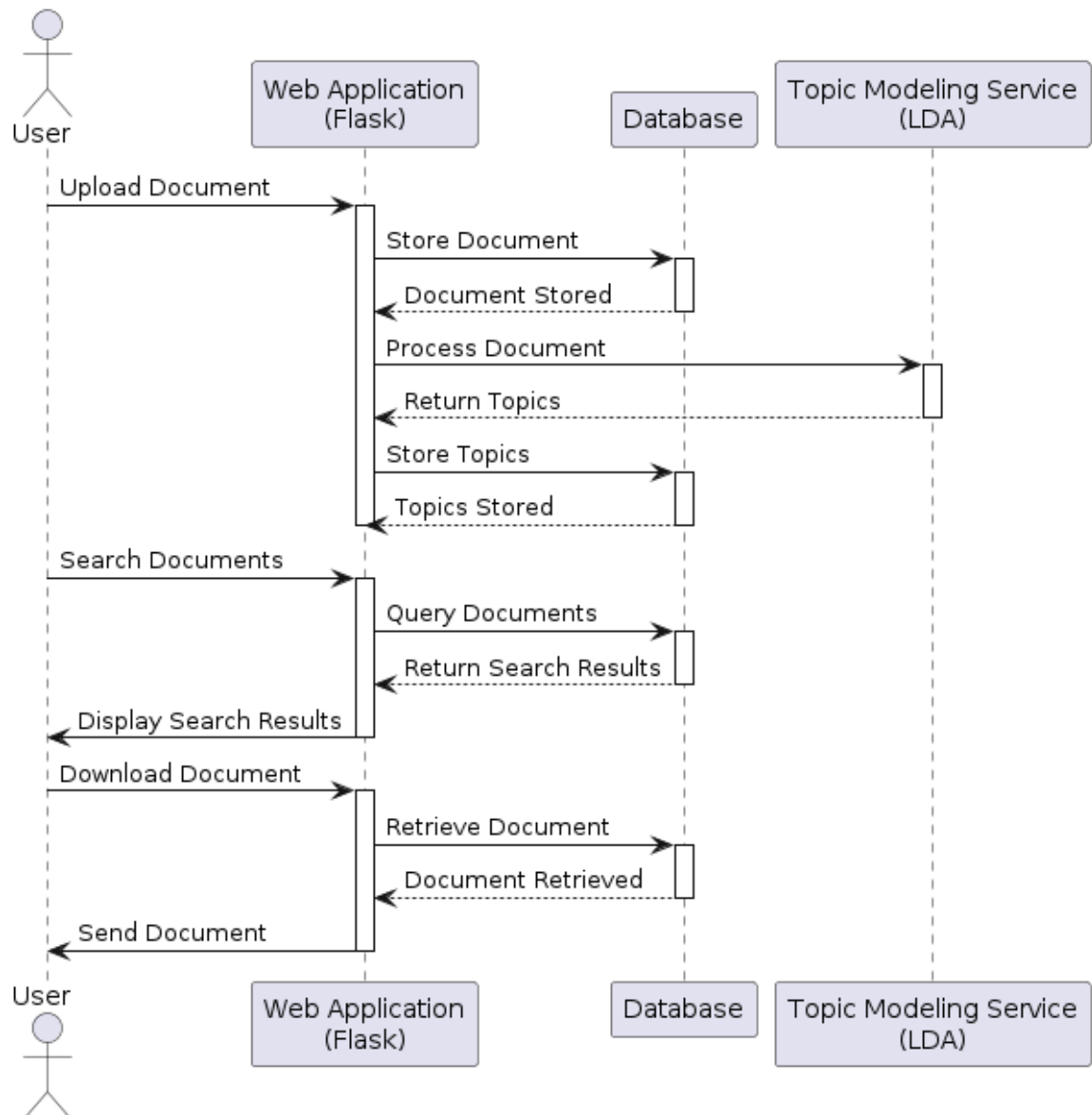


**Figure 3.3: Activity diagram of the proposed system**

### 3.6.2.2.1 Sequence diagram

Sequence Diagrams are interaction diagrams that provide a detailed account of how operations are carried out. They capture the interplay between objects in the context of a collaboration. Notably, Sequence Diagrams are time-focused, utilising the vertical axis of the

diagram to represent the chronological order of interactions and the exchange of messages. Sequence diagrams offer a powerful tool for understanding the dynamic behaviour of complex systems. These diagrams provide valuable insights into the system's inner workings and dynamic behaviour by visually depicting the interactions between components over time.

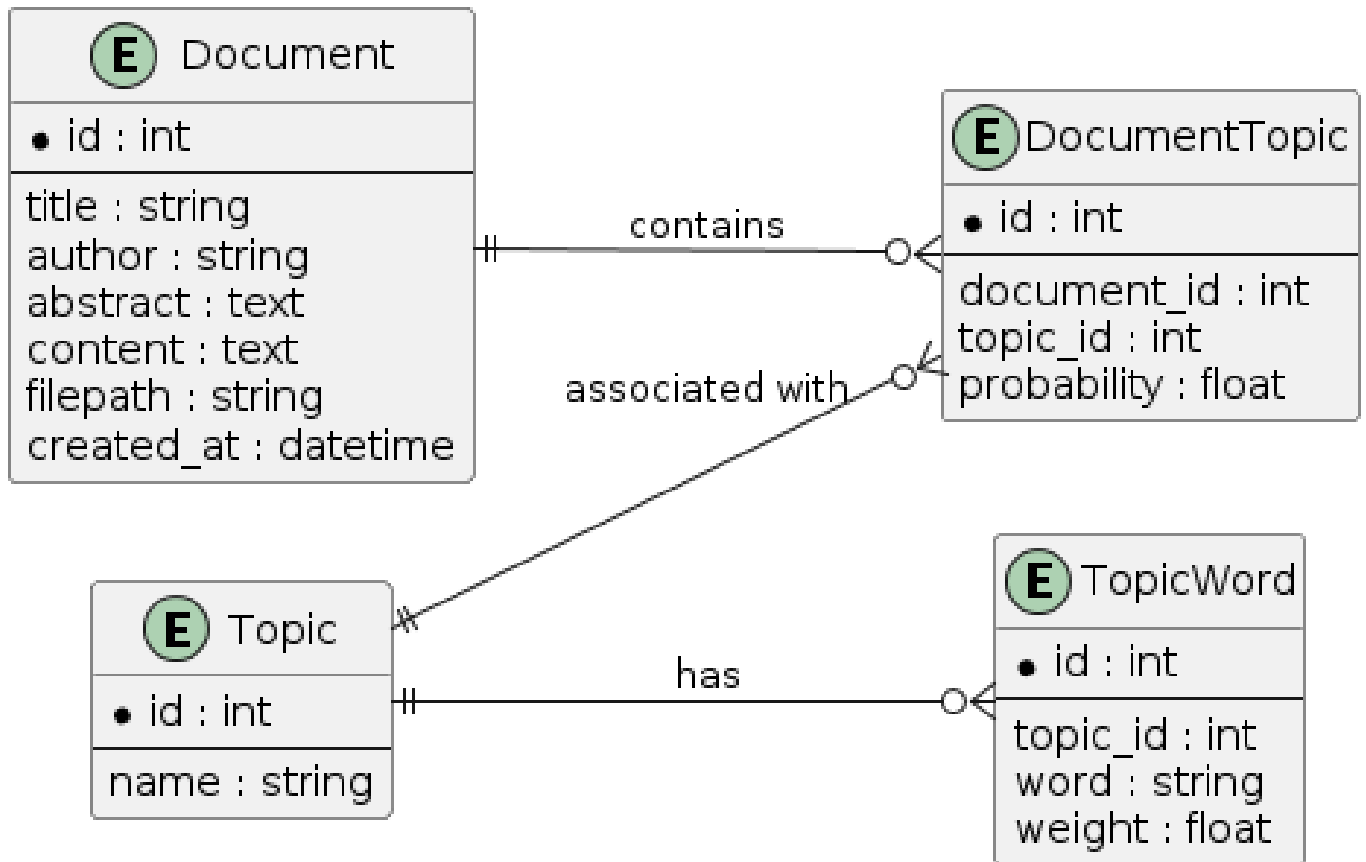


**Figure 3.4: Sequence diagram of the proposed system**

### C. Conceptual design

This section presents a comprehensive visual analysis of the proposed system's data structure, featuring the Entity Relationship (ER) diagram and corresponding database tables. Figure 3.5 provides a detailed graphical representation of the system's data entities, attrib-

utes, and interrelationships, offering a clear and concise understanding of the database architecture and its underlying organisational logic.



**Figure 3.5: Entity relationship diagram of the proposed system**

### 3.7 Algorithm Design

This section presents the detailed algorithm design for implementing LDA, which includes the initialization of parameters, iterative updates, and the computation of document-topic and topic-word distributions. The design aims to ensure that the algorithm efficiently processes the document collection to generate meaningful and interpretable topics, facilitating effective information retrieval for users.

---

**Algorithm 3.1: LDA**

---

**Input:**  $D, K, \alpha, \beta$

**Initialize:**  $\theta, \phi, z$

**For each document**  $d$  **in**  $D$ :

**For each word**  $w$  **in** document  $d$ :

$z[d, w] \leftarrow \text{random}(1, K)$

**End For**

**End For**

$\text{bool} \leftarrow \text{false}$

**while not**  $\text{bool}$ :

**For each document**  $d$  **in**  $D$ :

**For each word**  $w$  **in** document  $d$ :

**For each topic**  $k$  **in**  $\text{range}(1, K+1)$ :

$p[k] \leftarrow (C\_WT[w, k] + \beta) / (\sum C\_WT[w, :] + W\beta) * (C\_DT[d, k] + \alpha) / (\sum C\_DT[d, :] + K\alpha)$

**End For**

$z[d, w] \leftarrow \text{sample}(p)$

**End For**

**End For**

$e\_temp \leftarrow \text{compute\_likelihood}(z, D, K)$

**if**  $e\_temp > e\_best$ :

$\text{bool} \leftarrow \text{true}$

$e\_best \leftarrow e\_temp$

**else:**

$e\_cur \leftarrow e\_temp$

**End If**

**End While**

**For each document**  $d$  **in**  $D$ :

**For each topic**  $k$  **in**  $\text{range}(1, K+1)$ :

$\theta[d, k] \leftarrow (C\_DT[d, k] + \alpha) / (\sum C\_DT[d, :] + K\alpha)$

**End For**

**End For**

**For each topic**  $k$  **in**  $\text{range}(1, K+1)$ :

**For each word**  $w$  **in**  $\text{range}(1, W+1)$ :

$\phi[k, w] \leftarrow (C\_WT[w, k] + \beta) / (\sum C\_WT[:, k] + W\beta)$

**End For**

**End For**

**Return**  $\theta, \phi, z$

---

### 3.8 Description of tables

The intelligent information retrieval system database design consists of some tables described in the sessions below.

#### 3.8.1 Document

The “Documents” table illustrated in Table 3.1 in the database structure contains information on the research paper uploaded, such as their title, author, abstract, the actual content, and the upload time.

**Table 3.1: Documents table**

Column name	Data type	Description
id	INT	Unique identifier
title	VARCHAR	Title of the document
author	VARCHAR	Author of the document
abstract	TEXT	Abstract of the document
content	LONGTEXT	Actual content of the document
file-path	VARCHAR	Path to the uploaded document file
created_at	TIMESTAMP	Date and time of document upload

#### 3.8.2 DocumentTopic

The “DocumentTopic” table illustrated in Table 3.2 contains the relationship between a topic and a document. Each document is mapped to a topic.

**Table 3.2: Documents table**

Column name	Data type	Description
id	INT	Unique identifier
document_id	INT	Foreign key referencing the document
topic_id	INT	Foreign key referencing the topic

#### 3.8.3 Topic

Table 3.3 displays the information on the topics identified by the LDA algorithm. Each topic is uniquely identified and can be associated with multiple documents

**Table 3.3: Documents table**

Column name	Data type	Description
id	INT	Unique identifier
name	VARCHAR	Label of the topic

### 3.8.4 TopicWord

The TopicWord table, as shown in 3.4 3.4, details the words connected to each topic. Each record in this table connects a topic with a particular word identified during the topic modelling process and the word's significance in the topic.

**Table 3.4: Documents table**

Column name	Data type	Description
id	INT	Unique identifier
topic_id	INT	Foreign key referencing topics
word	VARCHAR	Words that make up a topic
weight	FLOAT	Weight of the word in a topic

## **CHAPTER FOUR**

### **SYSTEM IMPLEMENTATION AND EVALUATION**

#### **4.1 Preamble**

This section provides a comprehensive overview of the implementation of the proposed intelligent information retrieval system. It outlines the hardware and software requirements for the system's operation, the methodology employed during its development, the various program modules and interfaces, and the evaluation of the system's performance.

#### **4.2 System Requirements**

The system requirements are the hardware and software constraints the system must satisfy for the system to be able to perform optimally.

##### **4.2.1 Hardware Requirements**

To operate at its best, a system must be equipped with hardware that meets or surpasses specific requirements, providing the necessary physical resources to support the software and execute tasks with maximum efficiency.

**Table 4.1: Hardware requirements table**

<b>S/N</b>	<b>REQUIREMENT</b>	<b>HARDWARE</b>
1	Processor	Intel Core i5 or higher / Apple M1 chip
2	Primary memory	16 GB RAM or higher
3	Architecture	64Bit (X64) for Windows / ARM X64 for macOS
4	Secondary storage	256GB SSD or higher

##### **4.2.2 Software Requirements**

To ensure the successful operation of the software system, the following requirements must be met. Table 4.2 represents the software requirements of the proposed system.

**Table 4.2: Software requirements table**

S/N	REQUIREMENT	SOFTWARE
1	Operating System	Mac OS: v11.0.0 or higher Windows: 10 or higher Linux: Ubuntu 20.04 or higher
2	Programming Language	Python
3	Database	MySQL
4	Topic Modeling Library	Gensim (for LDA implementation)
5	User Interface	HTML CSS and Bootstrap
6	Development Tool	Visual Studio Code

### 4.3 Implementation tools

The software project was built using these tools, which are crucial in successfully developing the intelligent information retrieval system.

#### 4.3.1 Python

Python is an expressive and flexible programming language that uniquely blends object orientation, interpretability, and high-level abstractions. Its built-in solid data structures and dynamic typing and binding capabilities make it an excellent quick prototyping and development option. The modular architecture of the language, facilitated by modules and packages, encourages code organisation and reuse, guaranteeing effective program development and maintenance. Python was used to build the system's backend, implement the latent Dirichlet allocation algorithm, and manage interaction with the database. Python has several useful libraries and frameworks, some of which were utilised in this project and include the following:

##### A. Genism

Gensim is an open-source library used for unsupervised topic modelling, document indexing, similarity-based retrieval, and other natural language processing tasks, employing contemporary statistical machine learning techniques. Gensim was used to perform topic modelling on the contents of the documents.

##### B. Flask

Flask is a compact and lightweight Python web framework that offers practical features and tools to facilitate the development of Python online applications. It is a more approachable framework that offers developers flexibility.

### **4.3.2 NLTK**

NLTK comprises libraries and software designed for symbolic and statistical processing of natural language in English using Python programming. NLTK was used in this project to preprocess documents uploaded by the user.

### **4.3.3 Visual Studio Code**

This project was made possible by the flexible open-source code editor Visual Studio Code, which offered a wide range of plugins and features that made it possible to design and integrate the frontend and backend components successfully.

### **4.3.4 MySQL**

MySQL is an open-source, relational database management system that stores and organizes data in a structured format, using rows and columns. The software allows users to develop, handle, and modify databases. It was utilised in this project to store information on the documents and topic modelling results.

## **4.4 Development Methodology**

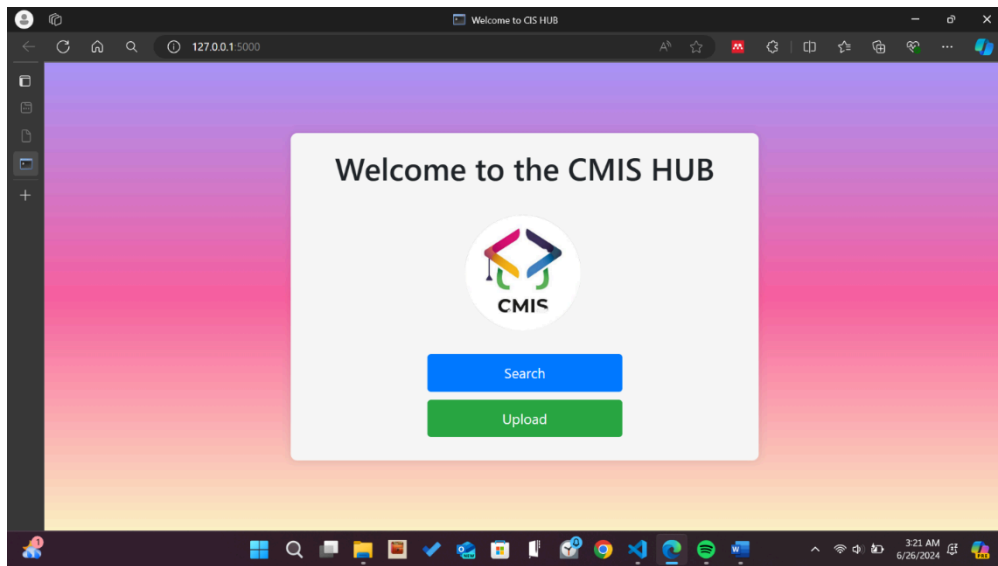
The software development methodology employed in this study is the agile methodology. The Agile methodology is a software development technique prioritizing cross-functional team cooperation, iterative development, and ongoing feedback. It divides the project into more manageable, smaller units called sprints, which usually span two to four weeks. It combines incremental and iterative process models and emphasizes the ability to adapt to processes and priorities customer satisfaction through the swift delivery of functional software products.

## **4.5 System Interfaces**

The system interface is essentially the channel via which the system user communicates with the system. The system's specified modules can process and exchange data and information more easily with its assistance. This section covers the various system interfaces of the intelligent information retrieval system.

### **4.5.1 The Home Page**

This contains the buttons to navigate to the search and upload module.



**Figure 4.1: Home page of system**

#### 4.5.2 The Upload Module

The user can upload his/her report here.

**Figure 4.2: Upload module**

### 4.5.3 The Search Page

The user can search for relevant documents here.

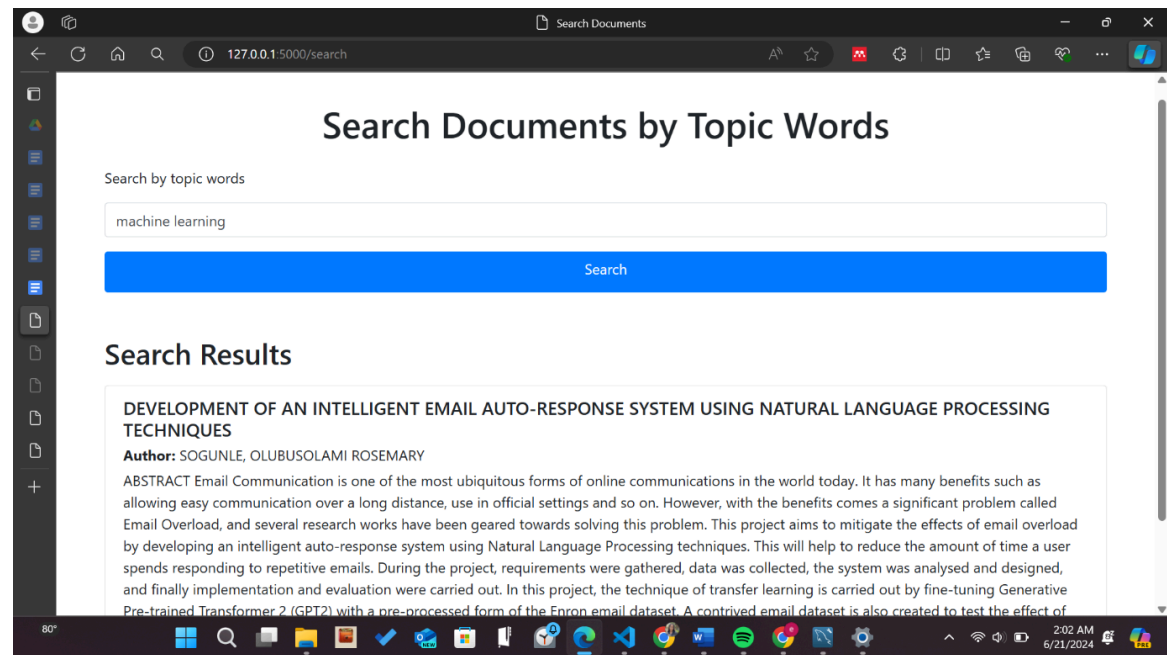


Figure 4.3: The search page

## 4.6 System Evaluation

Two primary evaluation criteria were adopted for the system evaluation. They include coherence and perplexity scores. Coherence is a way to measure how closely related the most important words are within a topic. Assessing coherence can assist in identifying the best number of topics for a specific dataset. At the same time, perplexity is widely used to assess the effectiveness of topic models such as LDA. It gauges the model's ability to predict documents that it has not previously encountered. A lower perplexity score suggests a higher level of model performance.

```
def evaluate_lda(lda_model, corpus, dictionary, documents):  
    coherence_model_lda = CoherenceModel(model=lda_model, texts=documents, dictionary=dictionary, coherence='c_v')  
    coherence_score = coherence_model_lda.get_coherence()  
  
    perplexity_score = lda_model.log_perplexity(corpus)  
  
    return coherence_score, perplexity_score
```

Figure 4.4: Coherence and perplexityscore

Evaluation Results	
Coherence Score	
0.3170661341412366	
Perplexity Score	
-7.361543661156962	
<a href="#">Back to Upload</a>	

**Figure 4.5: Evaluation results**

### 4.6.1 Discussion

#### A. Coherence score

Coherence scores typically range from 0 to 1, where scores closer to 1 indicate better coherence. A score of 0.317 suggests that the topics have a moderate level of interpretability and semantic similarity. Improvements might be necessary to enhance the coherence of the topics.

#### B. Perplexity score

Perplexity scores are usually negative, and lower (more negative) scores indicate better model performance. A score of  $-7.1$  suggests that the model is reasonably fitting the data, though there may still be room for improvement.

### 4.7 Usability testing

To evaluate the usability of the developed web application, a usability test was conducted using Google Forms. Participants were asked to perform specific tasks within the application and provide feedback on their experience. The survey aimed to gather insights on the ease of use, functionality, and overall user satisfaction with the system. Below is a summary of the results from the usability test.

How would you rate the speed and performance of the document upload and retrieval process?  
11 responses

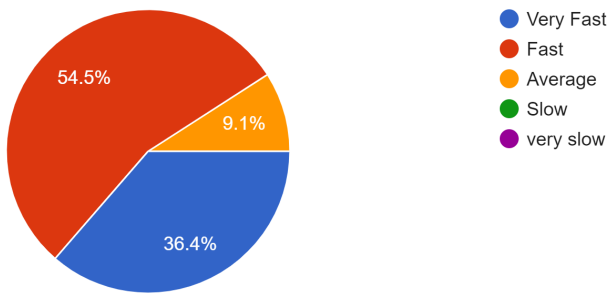


Figure 4.6: User question 1

How easy was it to upload a document on the platform? Untitled Question  
12 responses

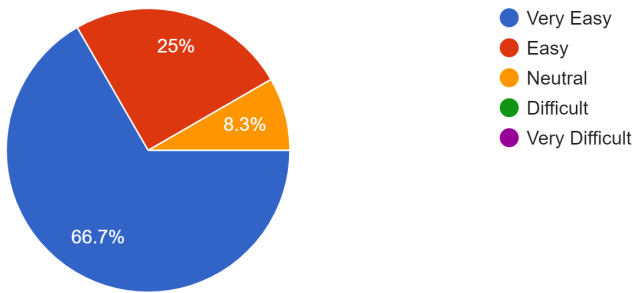


Figure 4.7: User question 2

Did the topic modeling feature meet your expectations in terms of identifying relevant topic words from the uploaded document?  
11 responses

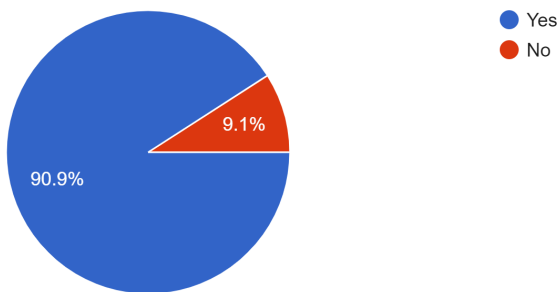


Figure 4.8: User question 3

The usability test for the web application showed that users were very satisfied with several important features. Most participants found it easy to upload documents, which suggests that the document upload interface is simple and intuitive to use. The topic modeling feature, which uses the Latent Dirichlet Allocation (LDA) algorithm, met the expectations of 90.9% of participants in terms of identifying relevant topic words from the uploaded documents, showcasing its effectiveness in extracting meaningful topics. Additionally, the ease of searching for and retrieving documents using topic words was well-received, with participants finding the search and retrieval functionality efficient and straightforward. Overall, the results highlight the system's effectiveness in facilitating document upload, topic modeling, and retrieval, with high user satisfaction.

## **CHAPTER FIVE**

### **CONCLUSION AND RECOMMENDATIONS**

#### **5.1 Summary**

This project focused on developing a web application that enables users to upload documents, perform topic modelling using LDA, and store the results in a MySQL database. The system facilitates users in searching for documents based on the topics identified by the LDA model, thereby enhancing the accessibility and relevance of the information. The application was built using Flask for both the front and back end, ensuring a seamless user experience. Throughout the project, challenges related to data preprocessing, algorithm optimisation, and system integration were addressed, resulting in a robust and efficient solution for document management and retrieval based on topic modelling.

#### **5.2 Recommendations**

- (i) Utilization of a Larger Dataset: Expanding the dataset with more diverse and comprehensive documents will improve the LDA model's ability to identify and generate meaningful topics, enhancing the overall accuracy and relevance of the recommendations.
- (ii) Enhancement of User Interface: Improving the user interface to be more intuitive and user-friendly will encourage greater user engagement and satisfaction.
- (iii) Incorporation of Advanced Algorithms: Exploring and integrating more advanced topic modelling algorithms, such as neural topic models, could provide better performance and deeper insights.

#### **5.3 conclusions**

The developed web application successfully demonstrates the utilisation of LDA for topic modelling to facilitate document retrieval based on thematic content. The system provides a practical and efficient solution for managing and searching large volumes of documents, rendering it a valuable tool for users seeking relevant information. While the project achieved its objectives, significant potential for further enhancements exists. By incorporating a larger dataset, improving the user interface, exploring advanced algorithms, implementing a feedback mechanism, and enhancing scalability, the system can be even more powerful and user-centric. This project lays a solid foundation for future content recommendation and information retrieval systems developments.

## REFERENCES

- Alkhodair, S. A., Fung, B. C., Rahman, O., & Hung, P. C. (2018). Improving interpretations of topic modeling in microblogs. *Journal of the Association for Information Science and Technology*, 69(4), 528–540. <https://doi.org/10.1002/asi.23980>
- Apte, C., Library., A. D., Data Mining., A. for Computing Machinery. Special Interest Group on Knowledge Discovery &, & Data., A. for Computing Machinery. Special Interest Group on Management of. (2011). *Proceedings of the 17th ACM SIGKDD international conference on Knowledge discovery and data mining*. (p. 1416–1417). ACM.
- Blei, D. M., Ng, A. Y., & Edu, J. B. (2003). *Latent Dirichlet Allocation* Michael I. Jordan (Vol. 3, pp. 993–1022).
- Boyd, C. (2018). What is latent semantic indexing why it won't help your seo. *Retrieved April, 20, 2019–2020*.
- Cheng, X., Yan, X., Lan, Y., & Guo, J. (2014). BTM: Topic modeling over short texts. *IEEE Transactions on Knowledge and Data Engineering*, 26(12), 2928–2941. <https://doi.org/10.1109/TKDE.2014.2313872>
- Chowdhury, G. G. (2003). *Natural language processing*. <http://eprints.cdlr.strath.ac.uk/2611/>
- Coroiu, A. M., & Nutu, M. N. (2019). *Topic Modeling in Medical Data Analysis. Case Study Based on Medical Records Analysis I st*.
- Dong, Hussain, H., Khadeer, F., Elizabeth, & Chang. (2008). *A survey in traditional information retrieval models*. 397–402.
- Foster, A., & Rafferty, P. (2011). *Innovations in information retrieval : perspectives for theory and practice* (p. 156–157). Facet Publishing.
- Gupta, R. K., Agarwalla, R., Naik, B. H., Evuri, J. R., Thapa, A., & Singh, T. D. (2022). Prediction of research trends using LDA based topic modeling. *Global Transitions Proceedings*, 3(1), 298–304. <https://doi.org/10.1016/j.gltp.2022.03.015>
- Hambarde, K. A., & Proenca, H. (2023). *Information Retrieval: Recent Advances and Beyond*. <https://doi.org/10.1109/ACCESS.2023.3295776>
- Harman, D. (2019). *Information retrieval: The early years* (Vol. 13, pp. 425–577). Now Publishers Inc. <https://doi.org/10.1561/15000000065>
- Hiemstra, D. (2009). *Information Retrieval Models \**.

- Islam, M. U. (2024). *What is an information retrieval system? And How Does It Work?*. <https://www.linkedin.com/pulse/what-information-retrieval-system-how-does-work-minhaj-ul-islam-vnklc/>
- Jelodar, H., Wang, Y., Yuan, C., Feng, X., Jiang, X., Li, Y., & Zhao, L. (2017). *Latent Dirichlet Allocation (LDA) and Topic modeling: models, applications, a survey*.
- Jelodar, H., Wang, Y., Yuan, C., Feng, X., Jiang, X., Li, Y., & Zhao, L. (2019). Latent Dirichlet allocation (LDA) and topic modeling: models, applications, a survey. *Multimedia Tools and Applications*, 78, 15169–15211.
- Kherwa, P., & Bansal, P. (2020). Topic Modeling: A Comprehensive Review. *EAI Endorsed Transactions on Scalable Information Systems*, 7(24), 1–16. <https://doi.org/10.4108/eai.13-7-2018.159623>
- Kochmar, E. (2022). *Getting Started with Natural Language Processing*. Simon, Schuster.
- Lal, N., Qamar, S., & Shiwani, S. (2016). Information Retrieval System and challenges with Dataspace. *International Journal of Computer Applications*, 147(8), 23–28. <https://doi.org/10.5120/ijca2016911128>
- Larson, R. R. (2011). *Information retrieval systems* (pp. 15–30). CRC Press. <https://doi.org/10.4018/ijtd.2018010101>
- Lashkari, A. H., Mahdavi, F., & Ghomi, V. (2009). *A boolean model in information retrieval for search engines*. 385–389. <https://doi.org/10.1109/ICIME.2009.101>
- Mahesh, B. (2018). Machine Learning Algorithms-A Review. *International Journal of Science and Research*. <https://doi.org/10.21275/ART20203995>
- Maklin, C. (2022). *Latent Dirichlet Allocation*.
- Pascual, F. (2019). *Introduction to Topic Modeling*.
- Peddireddi, Y. (2021). *Topic Modelling | Topic Modelling in Natural Language Processing*. <https://www.analyticsvidhya.com/blog/2021/05/topic-modelling-in-natural-language-processing/>
- Ramishamukhtar. (2023). *Boolean Retrieval Model*. <https://medium.com/@ramishamukhtar786/boolean-retrieval-model-be5843ae3c9e>
- Ribeiro, B. A. N., & Muntz, R. (1996). *A Belief Network Model for IR*.
- Roshdi, A., & Roohparvar, A. (2015). *Review: Information Retrieval Techniques and Applications* (Vol. 3, pp. 373–377).
- Sanderson, M., & Croft, W. B. (2012). *The history of information retrieval research*. 100, 1444–1451. <https://doi.org/10.1109/JPROC.2012.2189916>

- Shade, K., Oludele, A., Frank, I., & Samuel, A. (2012). Information Retrieval: An Overview. *International Journal of Advanced Research in Computer Science*, 3(5).
- Sharma, C., Sharma, S., & Sakshi. (2022). Latent DIRICHLET allocation (LDA) based information modelling on BLOCKCHAIN technology: a review of trends and research patterns used in integration. *Multimedia Tools and Applications*, 81(25), 36805–36831. <https://doi.org/10.1007/s11042-022-13500-z>
- Singh, V. K. (2022). *VECTOR SPACE MODEL: AN INFORMATION RETRIEVAL SYSTEM*. <https://www.researchgate.net/publication/362060638>
- Sudheer, M. (2022). *Models for IR in NLP*. <https://www.scaler.com/topics/nlp/nlp-ir-models/>
- Witter, D. I., & Berry, M. W. (1998). DOWNDATING the latent semantic indexing model for conceptual information retrieval. *The Computer Journal*, 41(8), 589–601.
- Yu, J., Mohan, S., Putthividhya, D., & Wong, W. K. (2014). *Latent dirichlet allocation based diversified retrieval for e-commerce search*. 463–472. <https://doi.org/10.1145/2556195.2556215>
- Zhao, F., Ren, X., Yang, S., Han, Q., Zhao, P., & Yang, X. (2020). *Latent Dirichlet Allocation Model Training with Differential Privacy*. <http://arxiv.org/abs/2010.04391>
- Zhou, L., & Zhang, D. (2003). *NLPIR: A Theoretical Framework for Applying Natural Language Processing to Information Retrieval*.