electronics

MDPI

Article

Leveraging Generative AI in Short Document Indexing

Sara Bouzid [1,*] and Loïs Piron [2]

[1] Complex Systems Modeling Laboratory, Cadi Ayyad University, Marrakesh 40000, Morocco
[2] Independent Researcher, 13120 Gardanne, France
[*] Correspondence: sara.bouzid@uca.ac.ma

Abstract: The efficiency of information retrieval systems primarily depends on the effective representation of documents during query processing. This representation is mainly constructed from relevant document terms identified and selected during their indexing, which are then used for retrieval. However, when documents contain only a few features, such as in short documents, the resulting representation may be information-poor due to a lack of index terms and their lack of relevance. Although document representation can be enriched using techniques like word embeddings, these techniques require large pre-trained datasets, which are often unavailable in the context of domain-specific short documents. This study investigates a new approach to enrich document representation during indexing using generative AI. In the proposed approach, relevant terms extracted from documents and preprocessed for indexing are enriched with a list of key terms suggested by a large language model (LLM). After conducting a small benchmark of several renowned LLM models for key term suggestions from a set of short texts, the GPT-4o model was chosen to experiment with the proposed indexing approach. The findings of this study yielded notable results, demonstrating that generative AI can efficiently fill the knowledge gap in document representation, regardless of the retrieval technique used.

Keywords: document indexing; short documents; generative AI; LLM; GPT; index term; information retrieval

check for updates

Citation: Bouzid, S.; Piron, L. Leveraging Generative AI in Short Document Indexing. *Electronics* **2024**, *13*, 3563. https://doi.org/10.3390/electronics13173563

Academic Editor: Arkaitz Zubiaga

Received: 5 August 2024
Revised: 3 September 2024
Accepted: 5 September 2024
Published: 8 September 2024

1. Introduction

Document indexing is the process of creating indexes for documents using a specific indexing scheme, which serves to accelerate document retrieval in search systems [1]. Indexing structures are organized by terms, which are mainly extracted from the documents themselves. These index terms are crucial in constructing document representations, typically in the form of term vectors, within information retrieval (IR) systems. The effectiveness of IR systems heavily relies on these document representations, commonly referred to as bag-of-words (BoW) [2], as they are matched with user queries by comparing their term vectors. When documents have minimal textual features and extensive figures, indexing and retrieving such documents becomes particularly challenging. Most existing approaches in the IR research field primarily focus on enhancing web-document and long-document retrieval [1,3–5] because there is a large amount of these types of resources available for experimentation, such as the TREC Corpora [6]. Short and domain-specific documents are less tackled by the scientific community due to the lack of availability and access to these resources. Yet, such resources are commonly used in business contexts for activity analysis, business process control, and decision-making. They are essential in fulfilling users' information needs. Examples of these resources include activity reporting documents, business workflow files, and accounting documents. We refer to these as short documents due to their minimal content. In related literature [7,8], short documents may include microblogs (e.g., X posts), short responses from work seminars, and abstracts. These types of short document often suffer from a limited number of terms, which can be insufficient to capture the full semantic meaning of the content they represent.

In fact, our problem statement directly arises from genuine needs identified during previous work in the industry [9]. Documents integral to core business operations typically feature figures, domain-specific terms, acronyms, and abbreviations. Unlike web and long documents, these documents lack a comprehensive basis for extracting meaningful features for document representation. In this context, automatic indexing may be ineffective, thereby impacting retrieval efficacy.

To date, studies related to IR have focused on enhancing the matching between user queries and documents [10–12]. The indexing stage in IR systems is often handled traditionally, involving the extraction and preprocessing of terms followed by the selection of those deemed relevant to the document's content. Relevance is typically calculated based on the frequency of terms within the document corpus or on the degree of their relationship with concepts from external knowledge resources [13,14].

Although automatic indexing can be performed traditionally on short and domain-specific documents, two main issues remain challenging to address: (1) how to obtain terms that best represent the content of documents, and (2) how to handle term specificity and scarcity. The first issue involves identifying terms that are most relevant to the subject matter of the documents. Recent studies propose using combined weighting schemes based on term frequency and semantic metrics [15–17] to identify relevant terms in documents to be indexed. However, term frequency may be useless for short documents, which contain only a few textual features. The second issue concerns obtaining accurate BoW representations of document content and user queries, even when the terms are rare or specific to a particular knowledge domain. This process requires handling information contextualizing these terms. Existing approaches to document indexing and retrieval address domain specificity and term rarity using external knowledge resources such as domain ontologies [5,18], thesauri [19,20] and lexical databases like WordNET [10,21,22]. However, these lexical resources suffer from limited coverage [21], particularly in specialized domains, and building knowledge resources from scratch may be complex and time-consuming. With the advent of artificial intelligence (AI), models using word embeddings [23] and deep learning techniques [24] have gained growing interest. These techniques have demonstrated their ability to capture nuanced semantic relationships between words [25,26].

In general, although significant progress has been made in document indexing through ontological approaches and term weighting schemes, many of these methods assume a variety and abundance of document terms. However, there is a notable gap in approaches addressing term scarcity and specificity, particularly in domain-specific short documents. These documents often suffer from information-poor features that are insufficient for effective indexing and retrieval.

In this study, we propose to use generative AI to produce relevant terms representing the content of documents for indexing purposes. Given the limited knowledge content in short documents, we assume that generative AI can identify additional relevant terms by leveraging pre-trained models that use advanced natural language processing and learning methods. The recent AI tools such as ChatGPT [27], Claude [28] and Gemini [29] show remarkable multitask capabilities in content analysis and creation, due to their extensive pre-training on diverse and vast datasets. The ability of these models to understand text topics can be harnessed to enhance IR tasks. Therefore, this paper investigates how to improve short-document indexing by leveraging generative AI, to enrich document index terms, traditionally extracted from documents, with additional relevant terms suggested by generative AI. To achieve this, we first conducted a comparative study of renown generative AI tools to identify the one best suited to capture key terms relevant to domain-specific short texts. The findings of this comparative study demonstrated the ability of the GPT-4o model to accurately suggest key terms regardless of the domain specificity of the given texts. In the next phase, we applied the GPT-4o model within our proposed indexing approach to a set of domain-specific short documents and compared the results with both the traditional indexing method and an automated subject indexing tool. The findings of this study revealed significant improvements in document retrieval using the index terms generated by our proposed approach.

The remainder of this paper is organized as follows: Section 2 provides background information and recent related work on document indexing, with a focus on domain-specific documents. Section 3 introduces the approach that leverages generative AI for document indexing and outlines the research steps followed to implement this approach. Section 4 presents the experimental results. Section 5 discusses the findings and validates the concept of the proposed approach. Finally, conclusions are presented in Section 6.

2. Background and Literature Review

2.1. The Indexing Task

The IR field has seen significant developments over the years. Two key aspects are fundamentally addressed in IR systems: the indexing process and the retrieval process. Indexing is the process of creating document indexes using an indexing scheme, such as inverted index, signature, or dense vector index [1]. This task requires selecting a list of terms from the documents themselves or from external resources to create index terms [30]. The retrieval process entails selecting relevant documents based on their representations (i.e., term vectors) that match user queries. A final step involves ranking the documents based on the results of the query matching. This paper focuses on document indexing, as it is a fundamental step in IR systems and central to the purpose of this work.

Prior to indexing, documents go through a preprocessing stage that includes techniques like tokenization [31], stopword removal, and lemmatization or stemming [32]. Tokenization is the process of dividing the text into individual words, known as tokens, using a list of separators to identify them. Stopword removal involves identifying and removing frequent words, like pronouns, prepositions, and conjunctions, that do not provide meaningful information and may bias the analysis of term relevance. Lemmatization is the process of grouping word variants into a single and common base form (e.g., retrieve, retrieved, retrieving, retrieval) [33]. One step in this process could be stemming, which consists in cutting or replacing prefixes and suffixes to obtain the root form of words. Following the preprocessing stage, term frequency (TF) techniques such as the standard term occurrence frequency, can be calculated to weight the occurrences of terms during the retrieval stage using measures like TF-IDF [34] and BM25 [35]. These measures help to create balanced term vectors of document BoW representations. For short documents, however, the use of TF techniques may be less effective, as the terms are few and rarely repeated.

After text preprocessing, document indexes are created and stored in a specific scheme, typically an inverted index structure for its fast retrieval capabilities. An inverted index consists of a collection of terms (words) and posting lists [1]. Each term in this structure points to the documents that contain it. These terms are referred to as index terms. The posting list records document identifiers, term frequencies, and other information such as term positions within the text. This data structure enables the matching of user queries with documents in an IR system using index terms. However, because a vocabulary mismatch can occur between document terms and query terms, IR studies continue to innovate to improve both indexing and retrieval tasks.

In general, the indexing of text collections can be conducted using automatic and semi-automatic processes. Two types of approaches are employed for (semi-) automatic indexing [30]: subject indexing and content-based indexing. In subject indexing, documents are indexed using terms from established knowledge organization systems (KOS), such as subject headings thesauri and universal classification systems. Such approaches are typically used in digital library applications. In content-based indexing, relevant and unique terms are extracted directly from the document content to be used as index terms. This study focuses on the content-based approach.

2.2. Content-Based Indexing and Document Representation in IR

Existing techniques for content-based indexing primarily focus on term selection from documents using weighted measures. However, only a limited number of studies have addressed the research area of document indexing in the recent years [36]. Most IR-related studies predominantly focus on document representation during retrieval, assuming that document indexes used to construct term vectors can be consistently enriched using external knowledge resources for domain-specific documents [5,10]. In the biomedical domain, the Medical Subject Headings (MeSH) thesaurus is commonly used to disambiguate domain-specific terms. In the study by Boukhari and Omri [15,37], the authors proposed to combine the Vector Space Model (VSM) and Description Logic (DL) to enhance biomedical document representation. The MeSH thesaurus was used to identify morphological variants and the most relevant concepts within documents. Gabsi et al. [38] introduced a semantic weighting scheme to disambiguate biomedical terms to build document representation. The approach determines the importance of relevant MeSH concepts in documents through term frequency and semantic similarities with unambiguous MeSH concepts. Other research in the biomedical domain employs Wikipedia as a knowledge base to address synonymy and polysemy or to leverage Wikipedia's interlanguage links to convert concept vectors between languages [39,40]. In [17], Aliwy et al. proposed using word sense disambiguation (WSD) and named entity recognition (NER) in both indexing and retrieval tasks for digital library management systems. During the indexing phase, a Part-of-Speech (POS) tagging process is first applied, followed by a combination of three NER techniques: rule-based chunking with filtering, conditional random fields (CRF), and bidirectional LSTM-CNN, employing a voting mechanism to annotate and index library content.

Some studies propose using concept clusters to create clustered indexes. These clusters are used to classify documents into homogeneous groups for easier searching [41] or to identify document relatedness to enhance search results [42]. In the legal field, Costa and Pedrosa [43] proposed the BoC-Th approach to generate concept clusters from word vectors, weighted by their semantic relevance within a legal-specific thesaurus. In [8], Kozlowski and Rybinski proposed a clustering algorithm SnSRC, to cluster short texts related to brainstorming seminars by comparing term sets using cosine similarity to identify sense frames. The resulting clusters provided a related representation of short texts.

Summarization is another technique used to enhance document indexing. The aim of the approach is to reduce document size without losing meaning, to speed up the indexing time, to optimize storage space using relevant terms from the summarization step [44,45]. However, this approach is more suitable for medium to long text documents.

Recent advances in NLP offer new techniques like word embeddings to understand semantic relationships between words, and the latest IR-related studies have leveraged these techniques for document retrieval [46,47]. Word embeddings capture the semantic meaning of user queries and documents by representing query and document terms as vectors in a continuous vector space. Models such as Word2Vec [48], GloVe [49], or BERT [50] that are trained on extensive text corpora to capture word semantics are typically used for this purpose. Some studies have developed new embedding models derived from state-of-the-art models for specific domains. Examples include specialized pre-trained BERT models such as BioBERT [51] for the biomedical domain, SPBERT for scientific literature [26], AMP-BERT [52] for antimicrobial peptide predictions, and COVID-Twitter-BERT [53] for COVID-19 content on Twitter. In [54], Dai and Callan explored the use of BERT's contextualized term representations for indexing purposes. They proposed a Deep Contextualized Term Weighting framework (DeepCT) based on BERT's representations to obtain context-aware term weights. These deep term weights can be used in an inverted index structure to identify the importance of terms in documents, enabling more efficient retrieval.

Although word embeddings have led to significant improvements in text representation, these techniques are primarily designed for retrieval tasks rather than indexing tasks. They require a minimum of consistent index terms and large pre-trained datasets to generate rich term vectors. In domain-specific IR studies, the lack of training data is often addressed by using specialized ontologies to index documents with machine learning techniques. In a recent study by Sharma and Kumar [16], the authors proposed a hybrid semantic document indexing approach that combines the Skip-gram model with negative sampling-based machine learning and a domain ontology to identify relevant concepts for indexing unstructured documents.

Current trends in semantic indexing encourage the use of AI to automatically index documents according to appropriate predefined concepts [30]. Examples of AI tools for automated subject indexing services include Annif [55] and Finto AI [56]. Annif uses a controlled vocabulary in English, Finnish, and Swedish, and a combination of NLP and machine learning techniques to suggest a list of terms for indexing from a given text or document. Finto AI is based on Annif API; however, authors limited the generated indexes to 20 terms. Today, the new generation of deep neural network models, such as transformers [50], have achieved outstanding results in training massive amounts of text data, leading to the rise of several large language models (LLMs) [57]. Examples include GPT [58], LaMDA [59], and LlaMA [60]. The GPT (Generative Pre-trained Transformer) model, one of the most popular LLMs, enables to predict new words or next sequences of words based on the context provided by the preceding words [50]. Recent research has begun to leverage the GPT model in fields like text classification and sentiment analysis [61–63]. In IR-related work, LLMs are specifically used in Retrieval-Augmented Generation (RAG) approaches [64,65]. The RAG technique combines the LLM capabilities with IR tasks to bypass re-training when input prompts are outside the LLM's data scope. Given a source like Wikipedia, the LLM uses the content of a set of documents as context with the original input prompt to generate specialized responses. This technique enables the LLM to access the most recent information and produce up-to-date outputs [66].

In summary, despite significant progress in IR research, most existing approaches assume the availability of a diverse and abundant set of document terms. However, there is a notable gap in addressing term scarcity, particularly in domain-specific short documents. Unlike web or long documents, short documents often lack sufficient information-rich features for effective indexing and retrieval. While recent IR techniques leverage rich term vectors, such as word embeddings, to represent documents, these methods still require a consistent initial representation of documents through their index terms, something that short documents often lack. This study aims to fill this gap by enriching document index terms with additional terms. In this context, because LLMs have demonstrated remarkable capabilities in capturing the informational context of a given text, we hypothesize that they can identify contextually relevant terms for short-document indexing, thereby augmenting traditional indexing methods. The central hypothesis of this research is that augmenting document index terms with LLM-generated terms will improve the retrieval accuracy of short documents. This study seeks to validate this hypothesis.

3. Materials and Methods

3.1. Motivation

Because domain-specific short documents suffer from term sparsity and lack of comprehensive information, using knowledge structures with weighted schemes or advanced NLP techniques like word embeddings and machine learning seems essential to enrich document representation. However, in specific knowledge domains, document corpora for training are often limited or missing, whereas machine learning methods need a large number of these training documents to be efficient [67]. The use of knowledge resources to support learning tasks also presents challenges in this case. On one hand, existing thesauri and subject headings are highly specialized. Each domain needs its own knowledge structure. On the other hand, domain-specific knowledge structures are challenging to build and maintain [68], particularly due to the complexity and time investment needed, as well as the need for specialized expertise.

Therefore, we propose to test a new technique for short-document indexing without the need of creating new knowledge structures. The main idea is to use generative AI based on an LLM to capture the key concepts representing the content of a document and use them as additional key terms for indexing. This approach serves as a form of data augmentation, enabling the effective indexing of documents with limited features. The resulting index terms combine concepts extracted from the documents and those generated by the AI. This approach is motivated by the advancements witnessed with the recent AI tools in understanding and creating consistent content in various domains. Indeed, LLMs that use transformers are well-trained on massive amounts of diverse text data from the internet and other sources, making them fine-tuned for tasks like contextual understanding in particular knowledge fields [58]. Such models are valuable resources for improving the indexing of short documents.

3.2. Research Proposal

Figure 1 illustrates our proposed approach that combines a generative AI with the traditional indexing technique to enhance the retrieval of short documents.

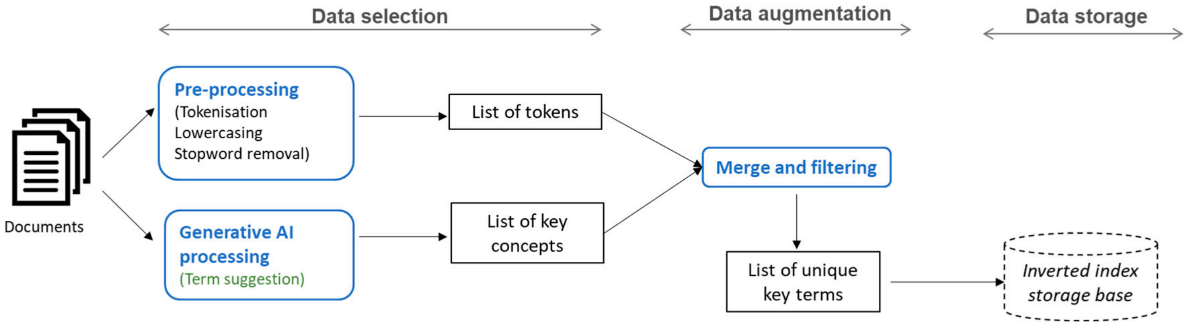


Figure 1. GenAI-based document indexing approach (Authors' proposal).

Basically, documents undergo preprocessing while simultaneously being analyzed by the generative AI. The preprocessing selects unique candidate terms from documents for indexing. The generative AI suggests a list of n terms. In the next step, the preprocessed candidate terms are merged with the AI-suggested terms to obtain a list of terms representing the content of each document. This list is then filtered to keep only unique terms, including compound words. It is important to note that we do not apply stemming and lemmatization during preprocessing, as the selected terms may be rare and specific to knowledge domains. The resulting terms are stored in an inverted index structure, commonly used in IR systems.

Overall, the proposed GenAI-based indexing approach follows three steps:

- Data selection: for each document, a list of tokens and concepts is selected in this step using traditional preprocessing tasks and a generative AI. Depending on the used LLM model, the suggested terms may differ from document terms. For this reason, we use both techniques to minimize any loss in the selected terms.
- Data augmentation: the resulting list of tokens from the preprocessing task is augmented with new terms suggested by the generative AI. Duplicated terms are removed. Compound words are considered unique terms.
- Data storage: the resulting list of index terms is stored in an inverted index structure.

It should be noted that this proposal does not rely entirely on generative AI, as the outcomes of such techniques can vary depending on the context provided, and these technologies are rapidly evolving. Generative AI is used only as a means to enrich document index terms. Moreover, this combined approach to document indexing ensures that generative AI, at worst, does not detract from the quality of terms traditionally extracted and, at best, enhances the index terms produced.

To implement and test the validity of this proposal, we took the following research steps:

- Selection of a generative AI: We conducted a benchmark of LLMs to evaluate their term suggestion capabilities, ultimately choosing the one best suited for our purpose.

- Selection of IR strategies: To assess the effectiveness of the index terms during document retrieval experiments, we chose two types of term vectors to perform the retrieval task: traditional vectors and enriched term vectors. These vectors were used to match query terms with document terms during experimentation.
- Data selection: This step consisted of selecting short documents in domain specific fields to apply the GenAI-based indexing approach.
- Experimentation for indexing and retrieval tasks: To validate the core concept of the proposal (i.e., enhancing traditional indexing with LLM-generated terms), we compared the performance of the GenAI-based indexing approach against other indexing methods.

3.3. Generative AI Selection

3.3.1. AI Tools' Characteristics

To identify the generative AI API that best fits our goal of indexing documents across multiple knowledge domains, we conducted a benchmark test of some leading AI tools to test their word suggestion abilities with domain-specific short texts. This step was necessary to assess the feasibility of the approach before conducting experiments on a larger dataset. The AI tools selected for this test were ChatGPT 3.5 [69], ChatGPT-4o [70], Gemini 1.5 Pro [29], and Claude 3.5 Sonnet [28]. The AI tools were selected based on their popularity and ratings on specialized websites [71,72]. ChatGPT is one of the most revolutionary AI tools developed by OpenAI, renowned for its remarkable content creation capabilities. Gemini (formerly Bard) is Google's equivalent of ChatGPT. Claude is a cutting-edge AI tool developed by Anthropic, offering impressive content generation features like ChatGPT and Gemini. Table 1 presents the general characteristics of the selected generative AI tools.

Table 1. Characteristics of the selected generative AI tools.

Features	**ChatGPT-3.5**	**ChatGPT-4o**	**Gemini 1.5 Pro**	**Claude 3.5 Sonnet**
LLM	GPT-3.5	GPT-4o	Google Gemini	Claude 3.5
Number of parameters	175 Billion [73]	>1 Trillion [73]	Unknown	Unknown
Context window size [1]	4096 tokens [69] (≈3154 words)	128,000 tokens [70]	128,000 tokens [2] [29]	200K tokens (≈150,000 words) [28]
Inputs	Text	Text, Images, documents (Word documents, Plain text files, PDFs, several source code file formats)	Text, Images	Text, Images, Documents (Word documents, Plain text files, PDFs, Spreadsheets, several source code file formats)
Knowledge cutoff	September 2021	October 2023	Early 2023	April 2024
API Access for free tier	API key generation (after sign up), limited number of requests and tokens per minute	API key generation (after sign up), limited number of requests and tokens per minute	API key generation (after sign up), pay-as-you-go pricing for usage	Need approval from the commercial team to obtain an API key, pay-as-you-go pricing for usage

[1] AI tools can process and generate a text based on text input up to a maximum number of tokens (words, characters. . .). [2] A limited version can go up to 1 million tokens [29].

Based on these characteristics, GPT-4o stands out with a very large number of parameters (over 1 trillion) [73] compared to the other models, which theoretically makes it more efficient and accurate. The LLM Parameters are numerical values (weights) learned and adjusted during training to capture the model's knowledge and capabilities. The more parameters a model has, the more data it can process with greater expressiveness. The latest version of Claude AI is also noteworthy, as it has interesting characteristics similar to ChatGPT-4. It also has a recent cutoff knowledge (April 2024), which is essential for generating up-to-date data. Gemini has also interesting characteristics with an important context window size compared to ChatGPT-3.5, but since only short texts were involved in our experiments, this aspect was not a limitation.

3.3.2. Benchmark for Key Term Suggestion

The experimentation of these tools involved providing a text (or a document when possible) directly into the chat platform of each AI tool and requesting a list of terms representing the provided textual content for indexing. We used 15 short texts (Appendix E), consisting of abstracts related to electronics, mathematics, and biomedicines, extracted from MDPI journals published in 2023. The number of words in the selected abstracts varies between 132 and 227. The choice of scientific abstracts was intentional to test the basic abilities of these tools to understand and suggest terms from well-written texts in specific knowledge domains. These abstracts were selected based on three criteria: the length of the text (it must be short), the specificity of the text (it must contain specific terms from a particular knowledge domain), and the understandability of the text (to assess the results). The main question asked was: "Can you suggest a list of key terms from this text (or document) for indexing purposes?". Starting from this question, several tests were conducted on the selected AI tools. Table 2 presents the tests carried out with their results. It should be noted that the generated key terms were assessed by comparing them to the keywords proposed by the authors in their articles, as well as by drawing on our expertise in these fields, with the contribution of three domain experts (one from each field).

Table 2. Tests of selected AI tools in key term suggestions.

Tests		ChatGPT-3.5	ChatGPT-4o	Gemini-1.5	Claude-3.5
Test 1: Request for n relevant terms	Number of suggested terms	15 to 20	20 to 25	11 to 18	16 to 25
	Number of shared terms	6 to 15			
	Relevancy [1]	99%	99%	99%	99%
Test 2: Request for n relevant terms with $n \geq 30$	Number of suggested terms	30 to 35	30 to 45	27 to 30	30 to 35
	Number of shared terms	15 to 21			
	Relevancy	99%	99%	97%	99%
Test 3: Request for n relevant terms with $n \geq 60$	Number of suggested terms	60 to 65	60 to 70	18 to 26	60
	Number of shared terms	9 to 23			
	Relevancy	99%	99%	98%	98%
Test 4: Request for n relevant terms with $n < 45$	Number of suggested terms	20 to 35	20 to 43	14 to 29	20 to 32
	Number of shared terms	8 to 14			
	Relevancy	99%	99%	99%	99%

[1] Relevance to the topic of the provided text (calculated based on the number of relevant suggested terms over all the suggested terms).

In the initial attempts (Appendices A–C), we did not set a limit on the number of terms suggested by the AI tools. By default, for all the short texts provided, the AI tools suggested between 11 and 25 words, depending on the length and content of the provided abstracts. GPT-4o provided the largest number of key terms (20 to 25) compared to Gemini and Claude. Regarding shared terms between all the AI tools, at most half of the suggested terms are shared, including identical single terms or approximate similar compound words (excluding synonyms). We also noticed that Claude and ChatGPT models shared the most key terms, with an average similarity of 96% (calculated over all the tests), even though the GPT models suggested more words overall. When we asked for more terms from the AI tools in Test 2, Test 3, and Test 4, ChatGPT models again suggested the largest number of words. Many repeated terms appeared in the suggested lists due to the use of several compound words differentiated by synonyms. Indeed, we observed that the more terms we requested, the more new synonyms and derived terms were introduced into the suggestions. In Test 3 and Test 4, Gemini did not understand the assignments despite rephrasing the request in several ways, which explains the lower word count in its suggested lists compared to other AI tools. The ChatGPT models and Claude provided coherent lists of terms in each test, but ChatGPT-4o was more accurate and exhaustive in word suggestion. In general, the number of shared terms was specifically high between Claude and the used GPT models, while Gemini shared fewer terms with them. This explains the low numbers of shared suggested terms between all the tools tested.

Regarding term relevancy, the suggested terms were mainly relevant to the content of the provided abstracts, with only a few marginal errors. For instance, in the following article "https://www.mdpi.com/2079-9292/12/5/1247 (accessed on 10 June 2024)", the authors used the term "COVID-19" to explain that there is a growing use of video conferencing since the pandemic period. All the AI tools included this term in their suggested list of words. However, the article is actually related to computer science and delves into issues and methodologies regarding data privacy and protection in video conferencing applications. Using COVID-19 as an index term in this context may be confusing and could lead to retrieval errors, especially for queries related to COVID-19 information, since the article is not about COVID-19 itself. In general, relevancy decreased slightly when we asked for more than 30 and 60 terms, respectively. Indeed, the more terms we requested, the greater the margin of error became.

Regarding the form of the suggested lists, the ChatGPT and Claude models presented the key terms in a similar enumerated list format. In most cases, the key terms followed their order of appearance in the text (Appendices A and B). Gemini's suggestions were more organized, presenting key terms in a categorized manner derived from the structure of the provided text. Nonetheless, many of the suggested terms were more topics than key terms (e.g., "Performance Compared to Baseline Methods", "Relationship between Resources and Corporate Revenue"). Furthermore, Gemini's suggestions occasionally included parentheses for additional details or explanations (e.g., "Physical Experiment (Elite 6Dof robot)", "Manual Detection Challenges (Discrepancies in Appearance)") (Figure A4 in Appendix C).

3.3.3. Conclusion of the Benchmark

The figures in Table 2 show that GPT-4o leads in term suggestion, with an average of 39.12 suggested terms across all tests, followed by GPT-3.5 with an average of 35 terms, Claude with an average of 34.75 terms, and finally, Gemini with an average of 21.62 suggested terms. In test 1, where the AI tools were not constrained by the number of suggestions, GPT-4o suggested an average of 22.5 terms, Claude suggested 20.5 terms, GPT-3.5 suggested 17.5 terms, and Gemini suggested 14.5 terms. When a minimum of 30 terms was imposed for word suggestion, as in test 2, both GPT 3.5 and Claude adhered closely to this threshold, each averaging 32.5 suggested terms. GPT-4o again exceeded expectations with an average of 37.5 suggested terms, while Gemini suggested an average of 28.5 terms. In test 3, where a higher minimum threshold was imposed ($\geq$60), GPT-4o met this target, suggesting an average of 65 suggested terms. GPT-3.5 followed with an average of 62.5 terms, and Claude suggested exactly 60 terms. Gemini struggled with this assignment, resulting in a significantly lower average of 22 suggested terms.

Regarding term relevancy, all LLMs demonstrated significant abilities in suggesting terms relevant to the provided texts. Specifically, the GPT models reached an average relevancy rate of 99%, followed by Claude at 98.75%, and finally Gemini at 98.25%.

Based on this benchmark, GPT-4o, GPT-3.5, and Claude excelled with consistent lists of suggested key terms at each test with satisfactory relevancy. We also noticed that the GPT models are better at adapting to word count constraints while preserving consistency in suggesting key terms. To conclude, GPT-4o has the best ratio number of suggestions/relevancy with a ratio of 39.12/99%. GPT-3.5 followed with a ratio of 35/99% and Claude with a ratio of 34.75/98.75%. Gemini had the lowest ratio at 21.62/98.25%. These findings naturally led us to choose the GPT-4o model for our GenAI-based indexing approach.

In addition to these statistics from the benchmark, empirical observations during the testing of the AI tools also supported our conclusions. For instance, although Gemini presented a coherent list of suggested words during the tests, the type and form of the proposed terms were not suitable for indexing purposes. In addition, some inconsistencies were noticed during the use of Gemini. The latter had difficulties understanding certain requests even when they were similar to previous ones (Figure A5 in Appendix D). It also displayed inconsistencies in counting the number of suggested words (Figure A6 in Appendix D). Furthermore, the tool sometimes changed the answering format without a specific request from the user (Figure A7 in Appendix D). As a result, although Gemini is an interesting generative AI tool with many features, the weaknesses observed during this benchmark indicate that it is not suitable for key term suggestion from given texts.

Regarding GPT models and Claude, no specific inconsistencies were observed. Nonetheless, we must point out that the free plan of Claude 3.5 Sonnet is restrictive when used with prompts as it allows a limited number of requests per day. A paid version is necessary to conduct experiments without restrictions and to use the API in source codes. Furthermore, obtaining an API key requires approval from the commercial team of Anthropic. The accessibility and affordability of the AI API are also important criteria to consider in selecting a generative AI.

In view of these conclusions, we decided to conduct the experimentation of the GenAI-based indexing approach with the GPT-4o model, as it demonstrated more conclusive results in this benchmark. The accessibility to the API was also an important factor in this choice, despite the rate limit (i.e., the number of tokens per minute), which is not significantly restrictive, given the short length of the texts used. In addition, GPT-4o, which is based on GPT-4, is the largest pre-trained model to date compared to GPT-3.5, leveraging more data and computation power [73]. This makes it well-suited for addressing the knowledge gap in domain-specific contexts.

3.4. IR Experimentation Strategies

To evaluate the consistency of the index terms for document retrieval regardless of the term-matching techniques, two IR strategies were adopted. Typically, during document retrieval, user queries consisting of a list of words are compared with document indexes. These representations, known as word vectors, are compared in the vector space using a similarity measure [74]. In this study, two types of word vectors are used in the experimentation: the traditional BoW representation, which relies on word vectors of each query and candidate documents, and word embeddings, where dense vectors of words are produced. In the context of short documents featuring a few words with some disparity, the Continuous Bag of Words (CBOW) [48] training model of Word2Vec appears as the most appropriate model to use in this case. Indeed, Word2Vec has two main models to learn word embeddings: CBOW and Skip-Gram. CBOW relies on predicting a target (middle) word given the context (surrounding) words. Skip-Gram employs the opposite approach. The order of words is not important in CBOW, as it uses a BoW representation, which aligns with the context of this study. The main goal behind using two types of word vectors is to analyze and compare the impact of index terms on document retrieval. The cosine similarity measure (Equation (1)) is used to compute the similarity between the produced word vectors.

$$\cos\left(\vec{a_i}, \vec{b_j}\right) = \frac{\sum_{r=1}^{n} a_{ir} b_{jr}}{\sqrt{\sum_{r=1}^{n} a_{ir}^2}\,\sqrt{\sum_{r=1}^{n} b_{jr}^2}} \tag{1}$$

3.5. Data Selection

To experiment with the approach, 9350 short documents were selected, related to three different themes: finance, COVID-19, and sports. These documents, mainly composed of a few words and many figures, were extracted in CSV format from Kaggle datasets [75].

The number of terms (excluding figures) in the extracted documents ranges from 5 to 52 words. Given that two strategies were adopted in the IR system for word vectors and since Word2Vec requires pre-trained data, it was necessary to split data into a training set and a testing set. In our context, 70% (~6550) of the extracted datasets were used for pre-training and 30% (~2800) for testing. The training set is used to train the CBOW model of Word2Vec, while the testing set is used to evaluate its performance during document retrieval. We also used this testing set for indexing.

4. Experimental Results

The approach was implemented using the open-source software platform Apache Lucene (version 9.5). The equipment used for this experimentation was a Core i7 CPU machine with a 64-bit Windows operating system and 16 GB of RAM. The short documents downloaded from Kaggle were stored within the Lucene storage system. Lucene uses index file formats to store document-related indexes using an inverted index structure, where an index contains a set of documents, each referenced by an integer document number. Multiple documents can share the same indexes since they may contain similar terms describing their content. For the generative AI component, the GPT4-o model was integrated into our Lucene-based experiment platform using a generated API key.

In the following, two types of experimental results are analyzed to evaluate the approach: indexing results and retrieval results. The indexing results include data related to the number of indexes generated using our GenAI-based indexing approach. These results are compared with those obtained from other indexing methods, specifically traditional indexing and FintoAI-based indexing, which is a specialized indexing tool. In the retrieval results, the indexes produced by these three approaches are utilized to evaluate the effectiveness of the generated index terms during document retrieval.

4.1. Indexing Results

Table 3 presents the indexing results using the GenAI-based indexing approach. These figures are compared with a traditional indexing approach (i.e., without data augmentation with a generative AI) in Table 4. For 2800 documents, 32,546 index terms (i.e., unique words) were suggested by GPT-4o, and 33,217 index terms were obtained in total after merge and filtering with terms directly extracted from the documents. Based on our benchmark, we chose to let the AI API decide for the number of suggested terms for each document, which also helped avoid the risk of generating out-of-context terms. This strategy resulted in an average of 22.41 terms per document.

Table 3. Results of the GenAI-based indexing approach.

	Results
Total number of index terms [1]	33,217
Number of unique suggested terms (with GPT-4o)	32,546
Average number of suggested terms (with GPT-4o) per document [2]	22.41

[1] This number refers to all unique terms in the inverted index base. [2] This number includes repeated terms when they are shared between documents.

Table 4. Results of the traditional indexing approach.

	Results
Total number of index terms [1]	12,052
Average number of index terms per document [2]	10.24

[1] This number refers to all unique terms in the inverted index base. [2] this number includes repeated terms when they are shared between documents.

In contrast, the traditional indexing approach, which relies only on document terms, produced 12,052 index terms with an average of 10.24 terms per document. Note that the number of terms per document may include terms shared between documents, as the terms selected/suggested for a document can also be relevant for others.

Overall, the GenAI-based approach yielded 175.6% of data augmentation in index terms compared to the traditional approach.

To further evaluate the GenAI-based indexing approach, we compared its results with Finto AI, a dedicated automatic indexing tool for subject indexing for textual documents. Finto AI uses a generic ontology covering multiple knowledge domains. It takes text data or text file formats as input and provides a list of suggested terms for indexing as output. However, the tool limits the number of word suggestions to 10, 15, and 20 words. For this comparison, we chose the maximum number of words for indexing, which is 20. As shown in Table 5, using Finto AI instead of GPT-4o resulted in 29,814 index terms, and after merging and filtering with unique terms extracted from the documents, the total increased to 30,953 index terms. These figures are significant and closely align with the results obtained from the GenAI-based indexing approach.

Table 5. Results of the FintoAI-based indexing approach.

	Results
Total number of index terms [1]	30,953
Number of unique suggested terms by FintoAI	29,814
Average number of index terms per document [2]	20

[1] This number refers to all unique terms in the inverted index base. [2] this number includes repeated terms when they are shared between documents.

4.2. Retrieval Results

To assess the effectiveness of the resulting indexes on document retrieval, we conducted experiments with traditional word vectors and with dense vectors using the CBOW model of Word2Vec [76], which was trained on the preprocessed training set to create word embeddings. The resulting index terms from the three indexing approaches—GenAI-based, FintoAI-based, and traditional—were used separately in the retrieval process for comparison purposes. We executed 18 queries across three themes: finance, COVID-19, and sports, with 6 queries related to each theme from the dataset. The retrieval results were evaluated using traditional IR metrics: precision (Equation (2)), recall (Equation (3)), and F1 score (Equation (4)). These metrics are calculated for each query. Tables 6 and 7 summarize these results, where the average (Av) precision, average recall, and average F1 score of the 18 queries are presented.

$$precison = \frac{\#correctResults}{\#totalFound} \tag{2}$$

$$recall = \frac{\#correctResults}{\#correctResults + \#missedResults} \tag{3}$$

$$F1 = 2 \times \frac{precision \times recall}{precision + recall} \tag{4}$$

The experimental findings highlight the clear advantages of the GenAI-based indexing approach over other methods. The index terms generated by this approach significantly enhanced precision and recall across all queries compared to the traditional approach, regardless of the retrieval technique used. These results underscore the significant influence of index terms on document retrieval. Figures 2 and 3 provide detailed retrieval results by document themes after applying the GenAI-based indexing approach. Figure 2 focuses on the findings related to the use of traditional BoW vectors, while Figure 3 focuses on the findings related to the use of word embeddings. We can notice that both strategies produced almost similar results. The most accurate results were observed in queries related to COVID-19, primarily because documents in this domain are more specific and share less common information with documents from other topics. Overall, the use of word embeddings did not systematically impact the retrieval results. With both retrieval strategies, the precision consistently exceeded 0.8, while recall hovered around 0.9. These similarities in the results confirm the consistency of the index terms used in the GenAI-based indexing approach.

Table 6. Retrieval results of 18 queries using traditional word vectors and index terms resulting from each indexing approach.

Indexing Approach	Av Precision	Av Recall	Av F1
Traditional	0.556	0.334	0.418
FintoAI-based	0.593	0.582	0.588
GenAI-based (GPT-4o)	0.884	0.928	0.905

Table 7. Retrieval results of 18 queries using word embeddings (Word2Vec) and index terms resulting from each indexing approach.

Indexing Approach	Av Precision	Av Recall	Av F1
Traditional	0.616	0.445	0.517
FintoAI-based	0.723	0.566	0.635
GenAI-based (GPT-4o)	0.895	0.915	0.905

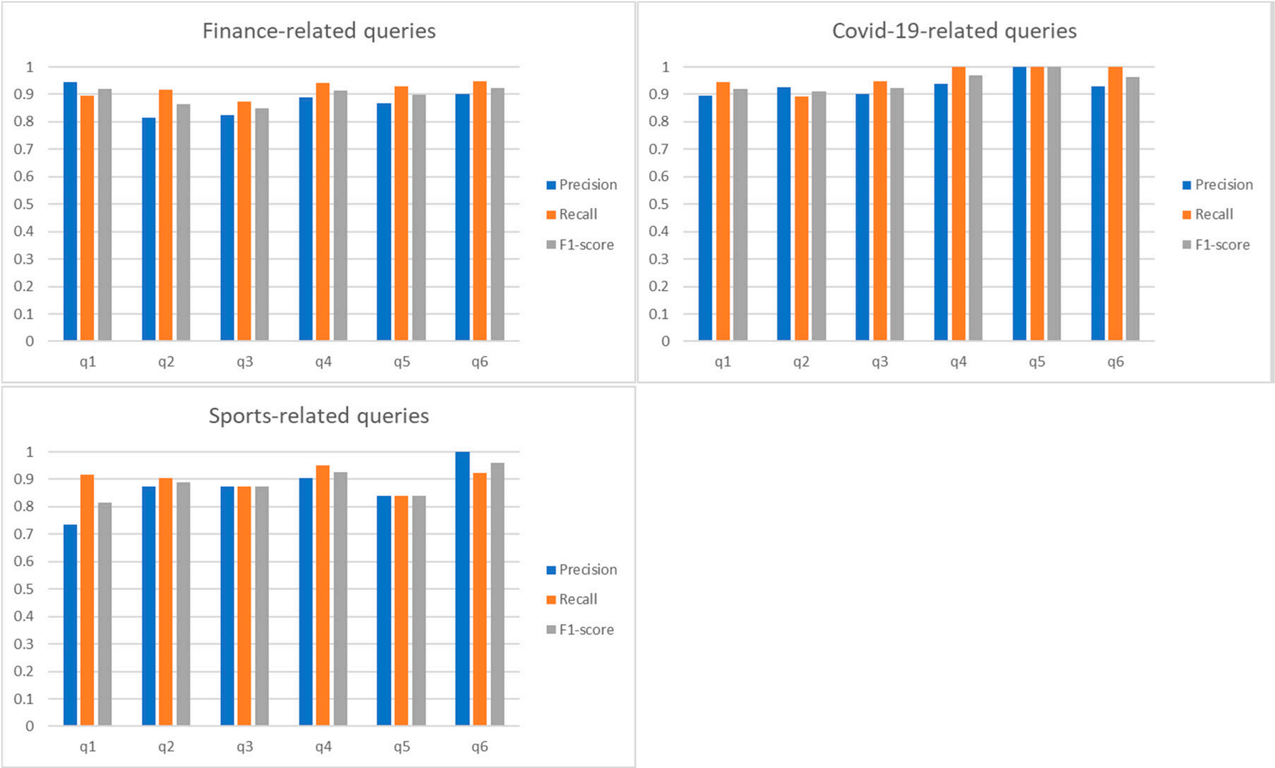


Figure 2. Detailed results by query themes using word vectors after applying the GenAI-based indexing approach.

Regarding the traditional indexing approach, its IR results are the least relevant among the other findings, with an average F1 score of 0.418. This outcome seems expected given that the documents were indexed using a poor-knowledge indexing approach. The use of word embeddings had an interesting impact on the retrieval results, with a 23.7% improvement in the average F1 score. This highlights the importance of having a rich document representation. Additionally, these results demonstrate that using word embeddings with short documents does not considerably improve their retrieval. Indeed, regardless of the indexing approach used, the impact of using dense vectors with Word2Vec was less significant than anticipated, especially with the GenAI-based approach. We can attribute this low impact to the specificity and scarcity of terms across documents, beside the lack of a large pre-trained dataset [77]. Word embeddings techniques require sufficient pre-trained data to learn word context and be effective. The more a model is pre-trained, the more significant its impact will be.

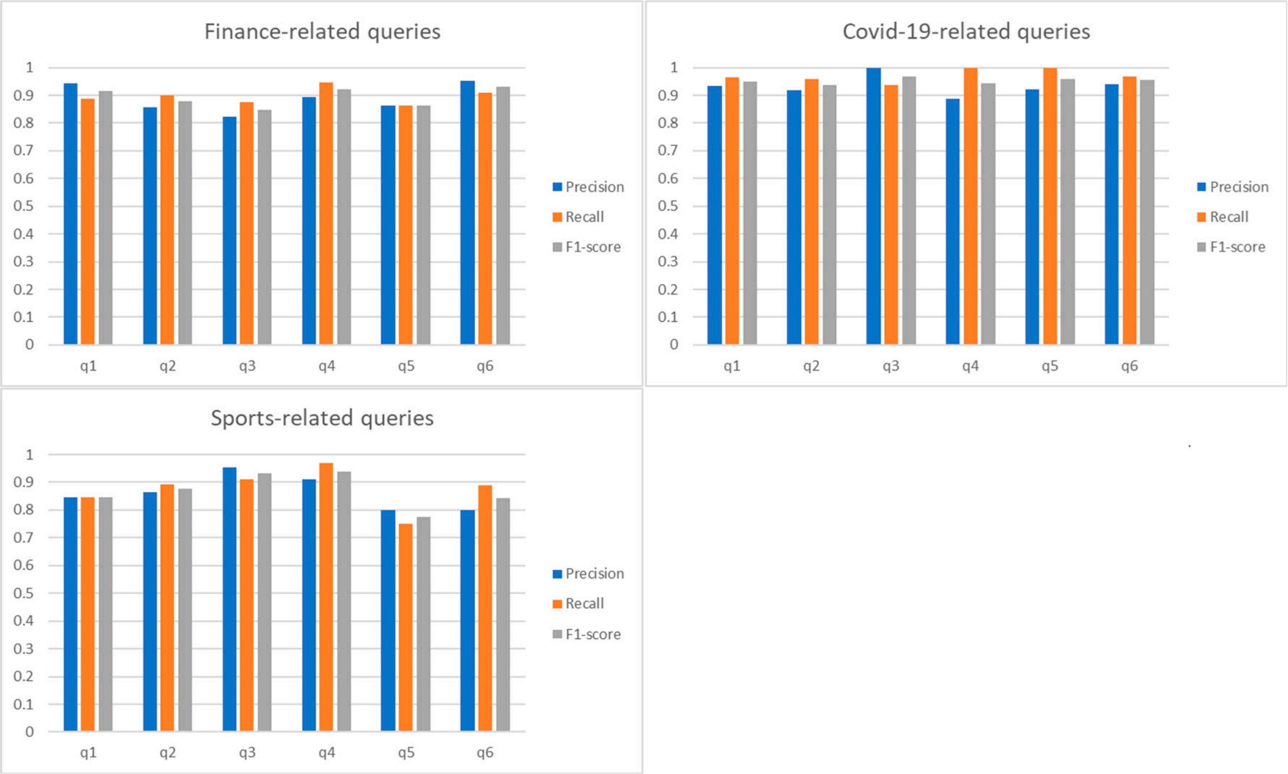


Figure 3. Detailed results by query themes using word embeddings (Word2Vec) after applying the GenAI-based indexing approach.

The FintoAI-based approach yielded a substantial number of index terms during the indexing process, comparable to the GenAI-based approach; however, the retrieval results were not significantly relevant, and in some cases, the results were irrelevant without the use of dense vectors. In fact, Finto AI is supported with a general ontology; therefore, using it as an automatic indexing service does not guarantee the generation of highly relevant index terms for specific documents. Consequently, these slight results indicate that the produced index terms with Finto AI are not entirely relevant for every document topic.

After the initial experiments, we added 2200 documents from the training set for indexing using the GenAI-based indexing approach, bringing the total number of indexed documents to 5000. Table 8 presents the detailed results of testing the same 18 queries. The resulting scores of the IR metrics remain consistent and they are comparable to the scores obtained in the initial tests, with an average F1 score of 0.808 for queries processed with traditional word vectors, and an average F1 score of 0.836 for queries processed with word embeddings.

Table 8. Retrieval results of 18 queries on 5000 indexed documents using traditional word vectors and word embeddings (Word2Vec).

Queries	Av Precision	Av Recall	Av F1
Finance-related queries	0.770	0.818	0.792
COVID-19-related queries	0.837	0.853	0.844
Sports-related queries	0.791	0.786	0.788
Finance-related queries using word embeddings	0.855	0.805	0.828
COVID-19-related queries using word embeddings	0.861	0.879	0.870
Sports-related queries using word embeddings	0.813	0.811	0.811

5. Discussion

This study aimed to test the hypothesis that augmenting document index terms with LLM-generated terms can improve the retrieval accuracy of short documents. The experimental results demonstrated that the GenAI-based indexing approach yielded an average of 22.41 index terms per document, significantly more than the 10.24 index terms produced by the traditional indexing approach. Overall, our approach generated 33,217 index terms, FintoAI-based indexing produced 30,953, and the traditional approach produced 12,052. These initial findings suggest that AI techniques can substantially increase the number of index terms. However, although both GenAI and FintoAI approaches produced more terms than the traditional approach, these results does not mean that the generated terms are consistently valuable for enhancing document retrieval. Thus, the retrieval experiments aimed to verify the effectiveness of the produced index terms. With both retrieval strategies—traditional term vectors and word embedding vectors –, the index terms from the GenAI-based indexing approach achieved a consistent average F1 score of 0.905. In comparison, the FintoAI-based indexing approach resulted in an average F1 score of 0.588 with traditional term vectors, which improved to 0.635 with word embeddings. The high F1 score obtained with the GenAI-based indexing indicates the superior relevance of the terms produced by our approach compared to those from the FintoAI-based indexing. The additional experiments, which involved indexing 5000 short documents using the GenAI-based indexing approach, also demonstrated consistently significant results. Queries processed with traditional word vectors yielded an average F1 score of 0.808, and those processed with word embeddings achieved an average F1 score of 0.836. These findings support the validity of our research hypothesis, confirming that the LLM-generated terms, used to augment traditional indexing, provide significant improvements in document retrieval. The validation of this hypothesis suggests that GPT-4o is well-suited for capturing significant terms from short texts, even within specific knowledge domains. With its extensive pre-training on large datasets, GPT-4o can identify terms that are commonly and frequently used in a given information context, making them relevant for indexing and retrieval. Recent studies [16,17] that have attempted to improve the indexing task in IR systems using weighted schemes supported with ontologies and machine learning have seen improvements in document retrieval accuracy of around 30%. In contrast, our approach achieved over a 75% improvement with both retrieval strategies—traditional term vectors and word embedding vectors. Nevertheless, it is important to note that this effectiveness depends on the GPT model maintaining up-to-date data. This limitation represents a key constraint of this study.

Ultimately, the findings from the experimentation validate the research hypothesis by demonstrating the capability of generative AI, particularly GPT-4o, to identify relevant key terms from texts that extend beyond those present in the documents themselves. This capability highlights GPT-4o as a practical solution for enhancing traditional indexing methods, especially when documents have limited textual content. By generating additional terms, generative AI can significantly improve IR systems by providing more comprehensive index terms. Furthermore, this data augmentation approach in indexing can address the gaps left by the absence of domain ontologies and thesauri, which are essential for improving

IR tasks in specific fields, as generative AI can produce additional terms contextualizing document content.

6. Conclusions

This paper studied a new approach to document indexing by leveraging LLMs like the GPT-4o model. The approach is specifically dedicated to short documents characterized by extensive figures and limited textual content, including domain-specific terms. Such documents present challenges in IR systems, from the indexing process to retrieval. While most IR-related studies focus on document retrieval techniques, this study focuses on the indexing process, arguing that effective indexing can significantly improve document retrieval. Applying generative AI in short-document indexing aims to bridge the information gap in document content by providing new index terms to each short document to be indexed. The first results of this study confirm the efficacy of using generative AI, particularly GPT-4o, one of the most advanced pre-trained models available. Furthermore, this experimentation demonstrated that it is possible to achieve significant advancements in IR tasks without relying on domain-specific ontologies and thesauri. The main limitation of this work is that LLMs are relatively recent technologies, and their features are still evolving along with their accessibility. These constraints can affect their sustainability when integrated into specific systems. Additionally, the cutoff knowledge of LLMs is a concern, as it is decisive to use the most up-to-date versions to ensure accuracy. Due to these limitations, our approach does not rely entirely on generative AI. Nevertheless, when a quick and easy-to-implement solution is needed, particularly for short-document retrieval, generative AI can be a practical means to support the indexing task, as demonstrated in this study.

Today, there is growing interest in using generative AI capabilities in specific research fields such as text classification and sentiment analysis [62,63]. This study reveals another interesting use of this technology to enhance IR systems. Based on our findings, future work will focus on two directions. The first is to explore the use of generative AI in short-document classification, with the aim of leveraging the resulting classifications to recommend documents with similar topics during retrieval tasks in IR systems. The second direction will focus on testing our GenAI-based indexing concept on domain-specific long documents to capture key concepts relevant for representing document topics in indexing. This approach could be particularly valuable in scientific literature [26], where identifying pertinent terms can be challenging, even when using term weighting schemes.

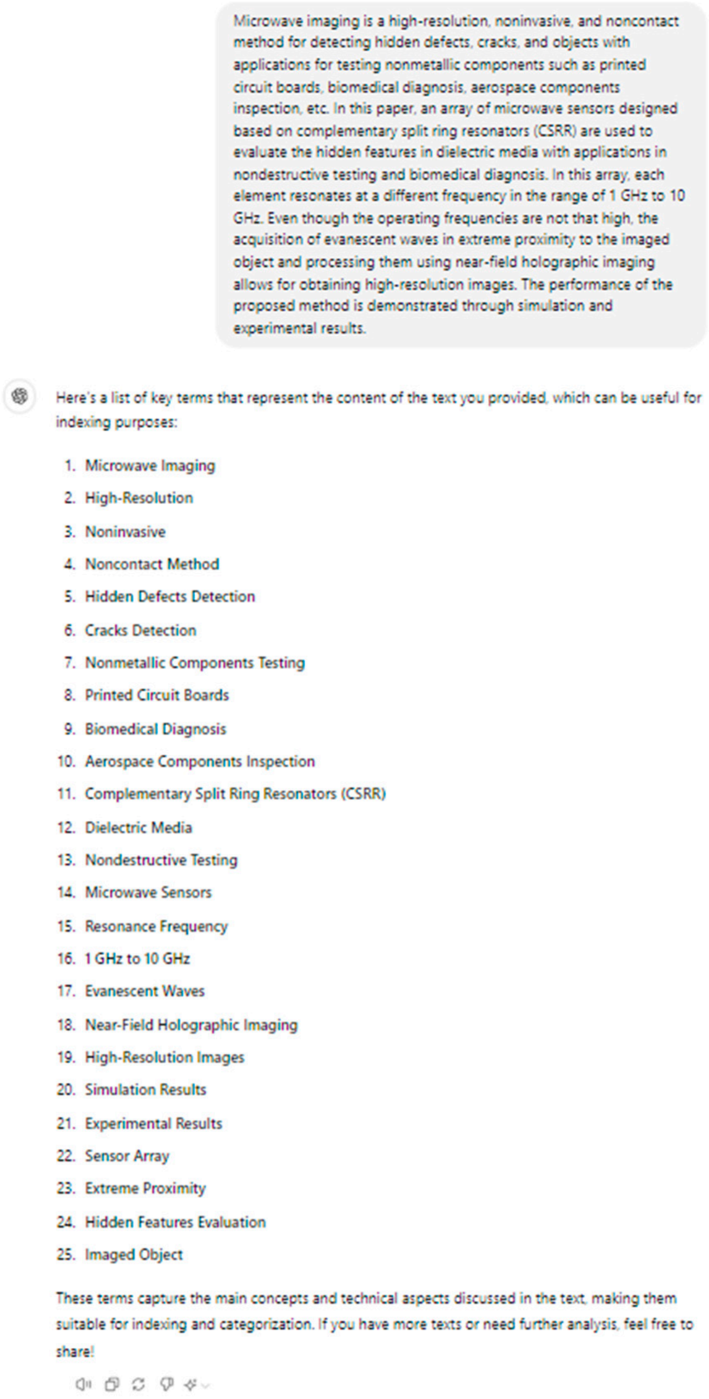
Author Contributions: Conceptualization, S.B.; methodology, S.B. and L.P.; Benchmark, S.B. and L.P.; implementation, S.B. and L.P.; validation, S.B. and L.P.; writing—original draft preparation, S.B.; writing—review and editing, S.B. and L.P. All authors have read and agreed to the published version of the manuscript.

Funding: This research received no external funding.

Data Availability Statement: The raw data supporting the conclusions of this article will be made available by the authors on request.

Conflicts of Interest: The authors declare no conflicts of interest.

Appendix A



Microwave imaging is a high-resolution, noninvasive, and noncontact method for detecting hidden defects, cracks, and objects with applications for testing nonmetallic components such as printed circuit boards, biomedical diagnosis, aerospace components inspection, etc. In this paper, an array of microwave sensors designed based on complementary split ring resonators (CSRR) are used to evaluate the hidden features in dielectric media with applications in nondestructive testing and biomedical diagnosis. In this array, each element resonates at a different frequency in the range of 1 GHz to 10 GHz. Even though the operating frequencies are not that high, the acquisition of evanescent waves in extreme proximity to the imaged object and processing them using near-field holographic imaging allows for obtaining high-resolution images. The performance of the proposed method is demonstrated through simulation and experimental results.

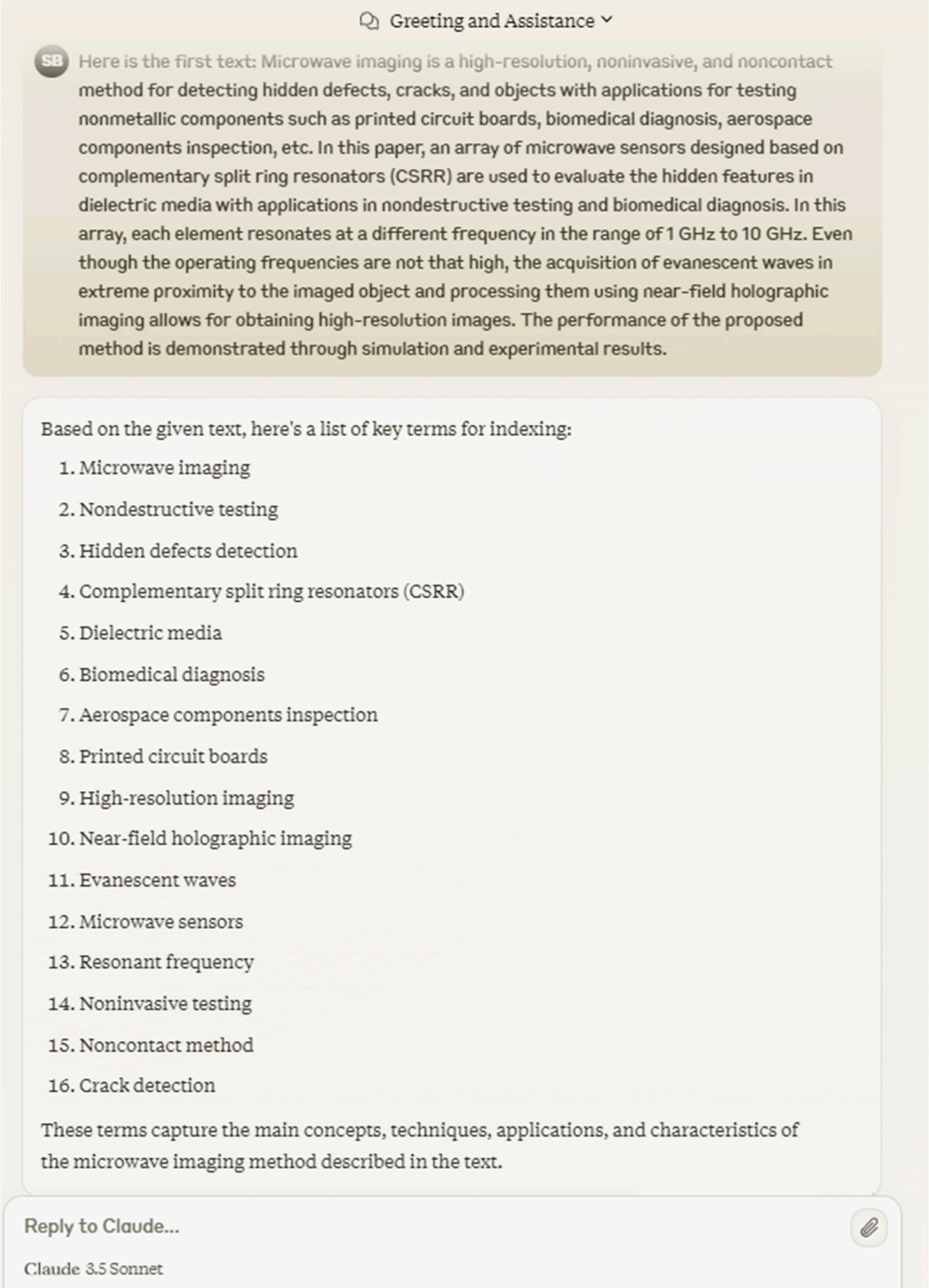
Here's a list of key terms that represent the content of the text you provided, which can be useful for indexing purposes:

1. Microwave Imaging
2. High-Resolution
3. Noninvasive
4. Noncontact Method
5. Hidden Defects Detection
6. Cracks Detection
7. Nonmetallic Components Testing
8. Printed Circuit Boards
9. Biomedical Diagnosis
10. Aerospace Components Inspection
11. Complementary Split Ring Resonators (CSRR)
12. Dielectric Media
13. Nondestructive Testing
14. Microwave Sensors
15. Resonance Frequency
16. 1 GHz to 10 GHz
17. Evanescent Waves
18. Near-Field Holographic Imaging
19. High-Resolution Images
20. Simulation Results
21. Experimental Results
22. Sensor Array
23. Extreme Proximity
24. Hidden Features Evaluation
25. Imaged Object

These terms capture the main concepts and technical aspects discussed in the text, making them suitable for indexing and categorization. If you have more texts or need further analysis, feel free to share!

Figure A1. Prompt for key term suggestion with ChatGPT-4o.

Appendix B



Here is the first text: Microwave imaging is a high-resolution, noninvasive, and noncontact method for detecting hidden defects, cracks, and objects with applications for testing nonmetallic components such as printed circuit boards, biomedical diagnosis, aerospace components inspection, etc. In this paper, an array of microwave sensors designed based on complementary split ring resonators (CSRR) are used to evaluate the hidden features in dielectric media with applications in nondestructive testing and biomedical diagnosis. In this array, each element resonates at a different frequency in the range of 1 GHz to 10 GHz. Even though the operating frequencies are not that high, the acquisition of evanescent waves in extreme proximity to the imaged object and processing them using near-field holographic imaging allows for obtaining high-resolution images. The performance of the proposed method is demonstrated through simulation and experimental results.

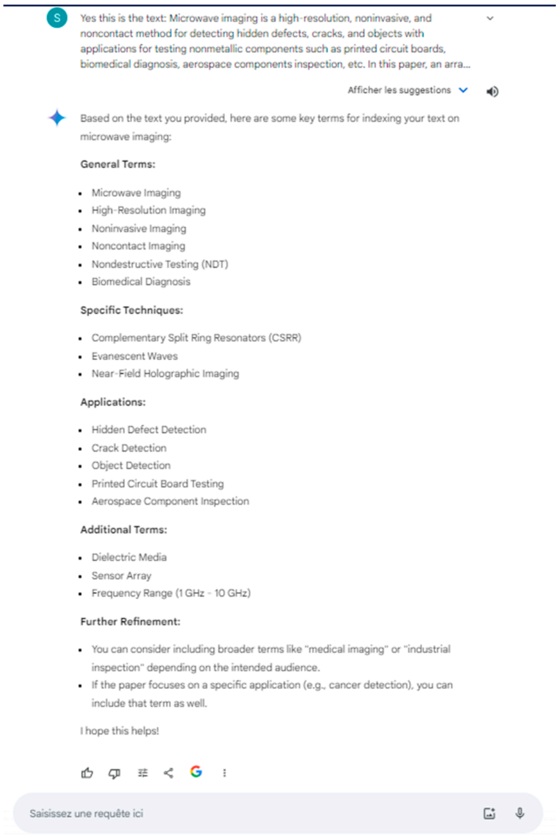
Based on the given text, here's a list of key terms for indexing:

1. Microwave imaging
2. Nondestructive testing
3. Hidden defects detection
4. Complementary split ring resonators (CSRR)
5. Dielectric media
6. Biomedical diagnosis
7. Aerospace components inspection
8. Printed circuit boards
9. High-resolution imaging
10. Near-field holographic imaging
11. Evanescent waves
12. Microwave sensors
13. Resonant frequency
14. Noninvasive testing
15. Noncontact method
16. Crack detection

These terms capture the main concepts, techniques, applications, and characteristics of the microwave imaging method described in the text.

Figure A2. Prompt for key term suggestion with Claude 3.5 Sonnet.

Appendix C



Yes this is the text: Microwave imaging is a high-resolution, noninvasive, and noncontact method for detecting hidden defects, cracks, and objects with applications for testing nonmetallic components such as printed circuit boards, biomedical diagnosis, aerospace components inspection, etc. In this paper, an arra...

Based on the text you provided, here are some key terms for indexing your text on microwave imaging:

General Terms:

- Microwave Imaging
- High-Resolution Imaging
- Noninvasive Imaging
- Noncontact Imaging
- Nondestructive Testing (NDT)
- Biomedical Diagnosis

Specific Techniques:

- Complementary Split Ring Resonators (CSRR)
- Evanescent Waves
- Near-Field Holographic Imaging

Applications:

- Hidden Defect Detection
- Crack Detection
- Object Detection
- Printed Circuit Board Testing
- Aerospace Component Inspection

Additional Terms:

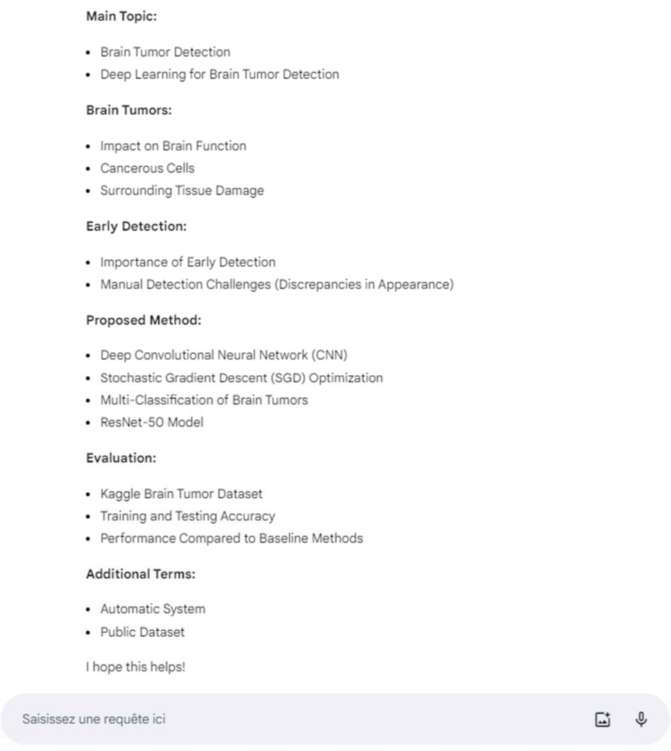
- Dielectric Media
- Sensor Array
- Frequency Range (1 GHz - 10 GHz)

Further Refinement:

- You can consider including broader terms like "medical imaging" or "industrial inspection" depending on the intended audience.
- If the paper focuses on a specific application (e.g., cancer detection), you can include that term as well.

I hope this helps!

Figure A3. Prompt for key term suggestion with Gemini 1.5.



Main Topic:

- Brain Tumor Detection
- Deep Learning for Brain Tumor Detection

Brain Tumors:

- Impact on Brain Function
- Cancerous Cells
- Surrounding Tissue Damage

Early Detection:

- Importance of Early Detection
- Manual Detection Challenges (Discrepancies in Appearance)

Proposed Method:

- Deep Convolutional Neural Network (CNN)
- Stochastic Gradient Descent (SGD) Optimization
- Multi-Classification of Brain Tumors
- ResNet-50 Model

Evaluation:

- Kaggle Brain Tumor Dataset
- Training and Testing Accuracy
- Performance Compared to Baseline Methods

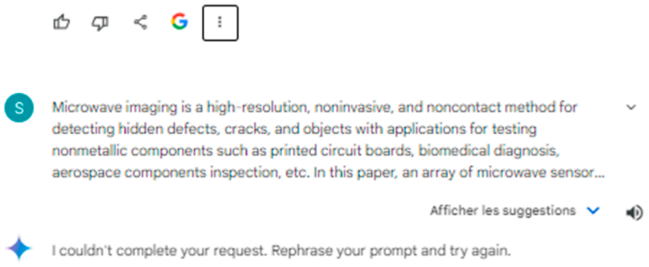
Additional Terms:

- Automatic System
- Public Dataset

I hope this helps!

Figure A4. Another list of categorized key terms suggested by Gemini. Some details are enclosed in parentheses.

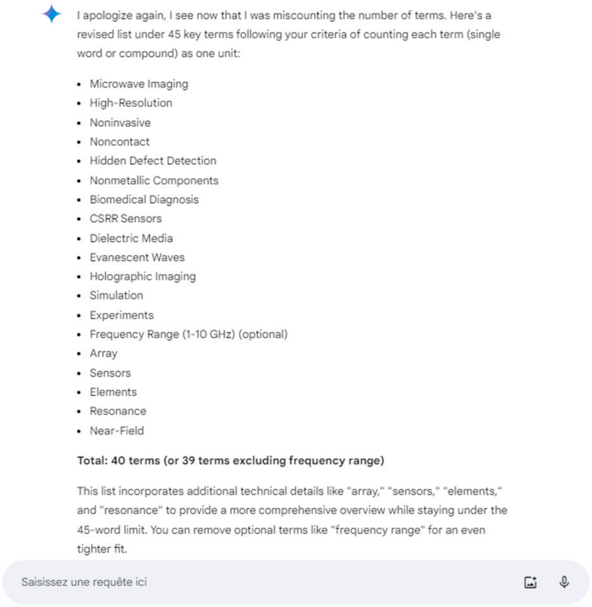
Appendix D



Microwave imaging is a high-resolution, noninvasive, and noncontact method for detecting hidden defects, cracks, and objects with applications for testing nonmetallic components such as printed circuit boards, biomedical diagnosis, aerospace components inspection, etc. In this paper, an array of microwave sensor...

I couldn't complete your request. Rephrase your prompt and try again.

Figure A5. Gemini's inability to respond to a request.



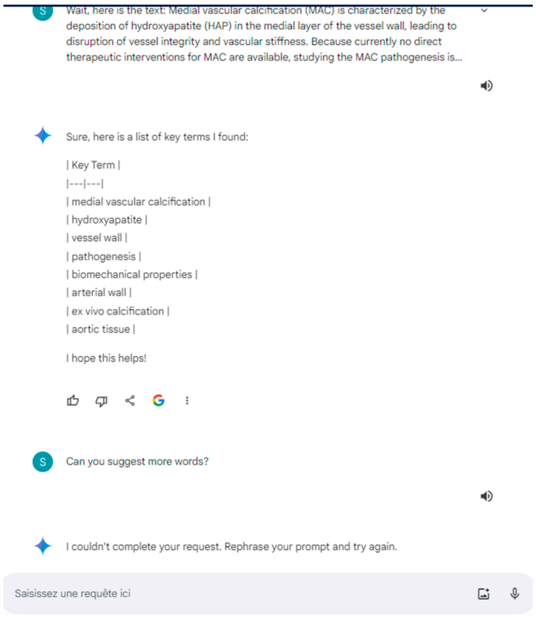
I apologize again, I see now that I was miscounting the number of terms. Here's a revised list under 45 key terms following your criteria of counting each term (single word or compound) as one unit:

- Microwave Imaging
- High-Resolution
- Noninvasive
- Noncontact
- Hidden Defect Detection
- Nonmetallic Components
- Biomedical Diagnosis
- CSRR Sensors
- Dielectric Media
- Evanescent Waves
- Holographic Imaging
- Simulation
- Experiments
- Frequency Range (1-10 GHz) (optional)
- Array
- Sensors
- Elements
- Resonance
- Near-Field

Total: 40 terms (or 39 terms excluding frequency range)

This list incorporates additional technical details like "array," "sensors," "elements," and "resonance" to provide a more comprehensive overview while staying under the 45-word limit. You can remove optional terms like "frequency range" for an even tighter fit.

Figure A6. Gemini's error in counting the number of suggested key terms.



Wait, here is the text: Medial vascular calcification (MAC) is characterized by the deposition of hydroxyapatite (HAP) in the medial layer of the vessel wall, leading to disruption of vessel integrity and vascular stiffness. Because currently no direct therapeutic interventions for MAC are available, studying the MAC pathogenesis is...

Sure, here is a list of key terms I found:

```
| Key Term |
|---|---|
| medial vascular calcification |
| hydroxyapatite |
| vessel wall |
| pathogenesis |
| biomechanical properties |
| arterial wall |
| ex vivo calcification |
| aortic tissue |
```

I hope this helps!

Can you suggest more words?

I couldn't complete your request. Rephrase your prompt and try again.

Figure A7. Another Gemini's suggested list of key terms related to test 4 (Table 4). A small number of suggested terms have been proposed with a new way of displaying the results.

Appendix E

Table A1. List of MDPI abstracts used in the Benchmark of generative AI tools.

Field	Number of Words	Link
Electronics	132	"https://www.mdpi.com/2079-9292/12/6/1507 (accessed on 10 June 2024)"
Electronics	184	"https://www.mdpi.com/2079-9292/12/6/1505 (accessed on 10 June 2024)"
Electronics	223	"https://www.mdpi.com/2079-9292/12/6/1502 (accessed on 10 June 2024)"
Electronics	200	"https://www.mdpi.com/2079-9292/12/5/1262 (accessed on 10 June 2024)"
Electronics	219	"https://www.mdpi.com/2079-9292/12/5/1247 (accessed on 10 June 2024)"
Biomedicines	140	"https://www.mdpi.com/2227-9059/11/1/211 (accessed on 10 June 2024)"
Biomedicines	140	"https://www.mdpi.com/2227-9059/11/1/208 (accessed on 10 June 2024)"
Biomedicines	211	"https://www.mdpi.com/2227-9059/11/1/193 (accessed on 10 June 2024)"
Biomedicines	224	"https://www.mdpi.com/2227-9059/11/1/189 (accessed on 10 June 2024)"
Biomedicines	203	"https://www.mdpi.com/2227-9059/11/1/184 (accessed on 10 June 2024)"
Mathematics	214	"https://www.mdpi.com/2227-7390/11/1/254 (accessed on 10 June 2024)"
Mathematics	189	"https://www.mdpi.com/2227-7390/11/1/245 (accessed on 10 June 2024)"
Mathematics	208	"https://www.mdpi.com/2227-7390/11/1/235 (accessed on 10 June 2024)"
Mathematics	227	"https://www.mdpi.com/2227-7390/11/3/783 (accessed on 10 June 2024)"
Mathematics	226	"https://www.mdpi.com/2227-7390/11/3/768 (accessed on 10 June 2024)"

References

1. Guo, J.; Cai, Y.; Fan, Y.; Sun, F.; Zhang, R.; Cheng, X. Semantic Models for the First-Stage Retrieval: A Comprehensive Review. *ACM Trans. Inf. Syst.* **2021**, *40*, 1–42. [CrossRef]
2. Carrillo, M.; Villatoro-Tello, E.; Lopez-Lopez, A.; Eliasmith, C.; Montes-y-Gomez, M.; Villasenõr-Pineda, L. Representing Context Information for Document Retrieval. In Proceedings of the International Conference on Flexible Query Answering Systems, Roskilde, Denmark, 26–28 October 2009; pp. 239–250.
3. Reddy, Y.V.B.; Reddy, S.N.; Reddy, S.S.S.N. Efficient Web-Information Retrieval Systems and Web Search Engines: A Survey. *Int. J. Mech. Eng. Technol.* **2017**, *25*, 123–125.
4. Tang, Y.; Zhang, R.; Guo, J.; Chen, J.; Zhu, Z.; Wang, S.; Yin, D.; Cheng, X. Semantic-Enhanced Differentiable Search Index Inspired by Learning Strategies. In Proceedings of the 29th ACM SIGKDD International Conference on Knowledge Discovery and Data Mining, Long Beach, CA, USA, 6–10 August 2023; pp. 4904–4913.
5. Asim, M.N.; Wasim, M.; Khan, M.U.G.; Mahmood, N.; Mahmood, W. The Use of Ontology in Retrieval: A Study on Textual, Multilingual, and Multimedia Retrieval. *IEEE Access* **2019**, *7*, 21662–21686. [CrossRef]
6. NIST TREC Data. Available online: https://trec.nist.gov/data.html (accessed on 20 July 2024).
7. Efron, M.; Organisciak, P.; Fenlon, K. Improving Retrieval of Short Texts through Document Expansion. In Proceedings of the International ACM SIGIR Conference on Research and Development in Information Retrieval, Portland, OR, USA, 12–16 August 2012; pp. 911–920.
8. Kozlowski, M.; Rybinski, H. Clustering of Semantically Enriched Short Texts. *J. Intell. Inf. Syst.* **2019**, *53*, 69–92. [CrossRef]
9. Bouzid, S. A Bottom-up Semantic Mapping Approach for Exploring Manufacturing Information Resources in Industry. *Comput. Syst. Sci. Eng.* **2017**, *32*, 243–256.
10. Jiang, Y. Semantically-Enhanced Information Retrieval Using Multiple Knowledge Sources. *Clust. Comput.* **2020**, *23*, 2925–2944. [CrossRef]
11. Tang, M.; Chen, J.; Chen, H.; Xu, Z.; Wang, Y.; Xie, M.; Lin, J. An Ontology-Improved Vector Space Model for Semantic Retrieval. *Electron. Libr.* **2020**, *38*, 919–942. [CrossRef]

12. Ormeño, P.; Mendoza, M.; Valle, C. Topic Models Ensembles for Ad-Hoc Information Retrieval. *Information* **2021**, *12*, 360. [CrossRef]
13. Yu, B. Research on Information Retrieval Model Based on Ontology. *EURASIP J. Wirel. Commun. Netw.* **2019**, *1*, 30. [CrossRef]
14. Jain, S.; Seeja, K.R.; Jindal, R. A Fuzzy Ontology Framework in Information Retrieval Using Semantic Query Expansion. *Int. J. Inf. Manag. Data Insights* **2021**, *1*, 100009. [CrossRef]
15. Boukhari, K.; Omri, M.N. DL-VSM Based Document Indexing Approach for Information Retrieval. *J. Ambient. Intell. Humaniz. Comput.* **2023**, *14*, 5383–5394. [CrossRef]
16. Sharma, A.; Kumar, S. Machine Learning and Ontology-Based Novel Semantic Document Indexing for Information Retrieval. *Comput. Ind. Eng.* **2023**, *176*, 108940. [CrossRef]
17. Aliwy, A.; Abbas, A.; Alkhayyat, A. NERWS: Towards Improving Information Retrieval of Digital Library Management System Using Named Entity Recognition and Word Sense. *Big Data Cogn. Comput.* **2021**, *5*, 59. [CrossRef]
18. Shakeri, M.; Sadeghi-Niaraki, A.; Choi, S.M.; AbuHmed, T. AR Search Engine: Semantic Information Retrieval for Augmented Reality Domain. *Sustainability* **2022**, *14*, 15681. [CrossRef]
19. Sunny, S.K.; Angadi, M. Evaluating the Effectiveness of Thesauri in Digital Information Retrieval Systems. *Electron. Libr.* **2018**, *36*, 55–70. [CrossRef]
20. Bedmar, I.S.; Martínez, P.; Martín, A.C. Search and Graph Database Technologies for Biomedical Semantic Indexing: Experimental Analysis. *JMIR Med. Inform.* **2017**, *5*, e7059. [CrossRef]
21. Hussain, M.J.; Bai, H.; Wasti, S.H.; Huang, G.; Jiang, Y. Evaluating Semantic Similarity and Relatedness between Concepts by Combining Taxonomic and Non-Taxonomic Semantic Features of WordNet and Wikipedia. *Inf. Sci.* **2023**, *625*, 673–699. [CrossRef]
22. Azad, H.K.; Deepak, A. A New Approach for Query Expansion Using Wikipedia and WordNet. *Inf. Sci.* **2019**, *492*, 147–163. [CrossRef]
23. Asudani, D.S.; Nagwani, N.K.; Singh, P. Impact of Word Embedding Models on Text Analytics in Deep Learning Environment: A Review. *Artif. Intell. Rev.* **2023**, *56*, 10345–10425. [CrossRef]
24. Ahmed, S.F.; Alam, M.S.B.i.n.; Hassan, M.; Rozbu, M.R.; Ishtiak, T.; Rafa, N.; Mofijur, M.; Shawkat Ali, A.B.M.; Gandomi, A.H. *Deep Learning Modelling Techniques: Current Progress, Applications, Advantages, and Challenges*; Springer: Dordrecht, The Netherlands, 2023; Volume 56, ISBN 0123456789.
25. Mhawi, D.N.; Oleiwi, H.W.; Saeed, N.H.; Al-Taie, H.L. An Efficient Information Retrieval System Using Evolutionary Algorithms. *Network* **2022**, *2*, 583–605. [CrossRef]
26. Wang, J.; Yang, Z.; Cheng, Z. Deep Pre-Training Transformers for Scientific Paper Representation. *Electronics* **2024**, *13*, 2123. [CrossRef]
27. Surden, H. Chatgpt, Ai Large Language Models, and Law. *Fordham Law Rev.* **2023**, *92*, 1941–1972.
28. Anthropic Claude 3.5 Sonnet. Available online: https://www.anthropic.com/news/claude-3-5-sonnet (accessed on 14 August 2024).
29. Pichai, S.; Hassabis, D. Our Next-Generation Model: Gemini 1.5. Available online: https://blog.google/technology/ai/google-gemini-next-generation-model-february-2024/#gemini-15 (accessed on 25 July 2024).
30. Golub, K. Automated Subject Indexing: An Overview. *Cat. Classif. Q.* **2021**, *59*, 702–719. [CrossRef]
31. Jurafsky, D.; Martin, J.H. *Speech and Language Processing: An Introduction to Natural Language Processing, Computational Linguistics, and Speech Recognition*, 1st ed.; Prentice Hall PTR: Hoboken, NJ, USA, 2000; ISBN 0130950696.
32. Singh, J.; Gupta, V. A Systematic Review of Text Stemming Techniques. *Artif. Intell. Rev.* **2017**, *48*, 157–217. [CrossRef]
33. Balakrishnan, V.; Humaidi, N.; Lloyd-Yemoh, E. Improving Document Relevancy Using Integrated Language Modeling Techniques. *Malays. J. Comput. Sci.* **2016**, *29*, 45–55. [CrossRef]
34. Salton, G.; Buckley, C. Term-Weighting Approaches in Automatic Text Retrieval. *Inf. Process. Manag.* **1988**, *24*, 513–523. [CrossRef]
35. Robertson, S.E.; Walker, S.; Beaulieu, M.M.; Gatford, M.; Payne, A. Okapi at TREC-4. In Proceedings of the 4th Text Retrieval Conference, Gaithersburg, MD, USA, 1–3 November 1995; pp. 73–97.
36. Desai, D.; Ghadge, A.; Wazare, R.; Bagade, J. A Comparative Study of Information Retrieval Models for Short Document Summaries. *Lect. Notes Data Eng. Commun. Technol.* **2022**, *75*, 547–562. [CrossRef]
37. Boukhari, K.; Omri, M.N. Approximate Matching-Based Unsupervised Document Indexing Approach: Application to Biomedical Domain. *Scientometrics* **2020**, *124*, 903–924. [CrossRef]
38. Gabsi, I.; Kammoun, H.; Souidi, D.; Amous, I. MeSH-Based Semantic Weighting Scheme to Enhance Document Indexing: Application on Biomedical Document Classification. *J. Inf. Knowl. Manag.* **2024**, 2450035. [CrossRef]
39. Mouriño García, M.A.; Pérez Rodríguez, R.; Anido Rifón, L. Wikipedia-Based Cross-Language Text Classification. *Inf. Sci.* **2017**, *406–407*, 12–28. [CrossRef]
40. Antonio Mouriño García, M.; Pérez Rodríguez, R.; Anido Rifón, L. Leveraging Wikipedia Knowledge to Classify Multilingual Biomedical Documents. *Artif. Intell. Med.* **2018**, *88*, 37–57. [CrossRef] [PubMed]
41. Chandwani, G.; Ahlawat, A.; Dubey, G. An Approach for Document Retrieval Using Cluster-Based Inverted Indexing. *J. Inf. Sci.* **2023**, *49*, 726–739. [CrossRef]
42. Inje, B.; Nagwanshi, K.K.; Rambola, R.K. An Efficient Document Information Retrieval Using Hybrid Global Search Optimization Algorithm with Density Based Clustering Technique. *Cluster Comput.* **2023**, *27*, 689–705. [CrossRef]

43. Costa, W.; Pedrosa, G.V. A Textual Representation Based on Bag-of-Concepts and Thesaurus for Legal Information Retrieval. In Proceedings of the Symposium on Knowledge Discovery, Mining and Learning, Brasilia, Brazil, 28 November–1 December 2022; pp. 114–121.
44. Ouadif, L.; El Ayachi, R.; Biniz, M. A New Approach of Documents Indexing Using Subject Modelling and Summarization. *J. Phys. Conf. Ser. Int. Conf. Math. Data Sci. (ICMDS)* **2020**, *1743*, 012032. [CrossRef]
45. Khalloufi, R.; El Ayachi, R.; Biniz, M.; Fakir, M.; Sarfraz, M. An Approach of Documents Indexing Using Summarization. In *Critical Approaches to Information Retrieval Research*; Sarfraz, M., Ed.; IGI Global: Hershey, PA, USA, 2020; pp. 78–86.
46. Bostan, S.; Bidoki, A.M.Z.; Pajoohan, M.-R. Improving Ranking Using Hybrid Custom Embedding Models on Persian Web. *J. Web Eng.* **2023**, *2*, 797–820. [CrossRef]
47. Gang, L.; Huanbin, Z.; Tongzhou, Z. Document Vector Representation with Enhanced Features Based on Doc2VecC. *Mob. Netw. Appl.* **2023**, 1–10. [CrossRef]
48. Mikolov, T.; Corrado, G.; Chen, K.; Dean, J. Efficient Estimation of Word Representations in Vector Space. *arXiv* **2013**, arXiv:1301.3781.
49. Pennington, J.; Socher, R.; Manning, C.D. GloVe: Global Vectors for Word Representation. In Proceedings of the 2014 Conference on Empirical Methods in Natural Language Processing (EMNLP), Doha, Qatar, 1 January 2014; pp. 1532–1543.
50. Devlin, J.; Chang, M.-W.; Lee, K.; Google, K.T.; Language, A.I. BERT: Pre-Training of Deep Bidirectional Transformers for Language Understanding. In Proceedings of the NAACL-HLT, Minneapolis, MN, USA, 2–7 June 2019; pp. 4171–4186.
51. Lee, J.; Yoon, W.; Kim, S.; Kim, D.; Kim, S.; So, C.H.; Kang, J. BioBERT: A Pre-Trained Biomedical Language Representation Model for Biomedical Text Mining. *Bioinformatics* **2020**, *36*, 1234–1240. [CrossRef]
52. Lee, H.; Lee, S.; Lee, I.; Nam, H. AMP-BERT: Prediction of Antimicrobial Peptide Function Based on a BERT Model. *Protein Sci.* **2023**, *32*, 1–13. [CrossRef]
53. Müller, M.; Salathé, M.; Kummervold, P.E. COVID-Twitter-BERT: A Natural Language Processing Model to Analyse COVID-19 Content on Twitter. *Front. Artif. Intell.* **2023**, *6*, 1023281. [CrossRef]
54. Dai, Z.; Callan, J. Context-Aware Term Weighting for First Stage Passage Retrieval. In Proceedings of the SIGIR '20: 43rd International ACM SIGIR Conference on Research and Development in Information Retrieval, Xi'an, China, 25–30 July 2020; pp. 1533–1536.
55. Suominen, O. Annif: DIY Automated Subject Indexing Using Multiple Algorithms. *Lib. Q. J. Assoc. Eur. Res. Libr.* **2019**, *29*, 1–25. [CrossRef]
56. Suominen, O.; Inkinen, J.; Lehtinen, M. Annif and Finto AI: Developing and Implementing Automated Subject Indexing. *JLIS.it* **2022**, *13*, 265–282. [CrossRef]
57. Liu, E.; Cui, C.; Zheng, K.; Neubig, G. Testing the Ability of Language Models to Interpret Figurative Language. In Proceedings of the 2022 Conference of the North American Chapter of the Association for Computational Linguistics: Human Language Technologies, Seattle, WA, USA, 10–15 July 2022; pp. 4437–4452.
58. Yenduri, G.; Ramalingam, M.; Selvi, G.C.; Supriya, Y.; Srivastava, G.; Maddikunta, P.K.R.; Raj, G.D.; Jhaveri, R.H.; Prabadevi, B.; Wang, W.; et al. GPT (Generative Pre-Trained Transformer)—A Comprehensive Review on Enabling Technologies, Potential Applications, Emerging Challenges, and Future Directions. *IEEE Access* **2024**, *12*, 54608–54649. [CrossRef]
59. Collins, E.; Ghahramani, Z. LaMDA: Our Breakthrough Conversation Technology. Available online: https://blog.google/technology/ai/lamda/ (accessed on 26 July 2024).
60. Meta Introducing Llama 3.1: Our Most Capable Models to Date. Available online: https://ai.meta.com/blog/meta-llama-3-1/ (accessed on 29 July 2024).
61. Wang, L.; Chen, R. Knowledge-Guided Prompt Learning for Few-Shot Text Classification. *Electronics* **2023**, *12*, 1486. [CrossRef]
62. Saleem, M.; Kim, J. Intent Aware Data Augmentation by Leveraging Generative AI for Stress Detection in Social Media Texts. *PeerJ Comput. Sci.* **2024**, *10*, 1–22. [CrossRef]
63. Alderazi, F.; Algosaibi, A.; Alabdullatif, M. Generative Artificial Intelligence in Topic- Sentiment Classification for Arabic Text: A Comparative Study with Possible Future Directions. *PeerJ Comput. Sci.* **2024**, *10*, 1–27. [CrossRef]
64. Lu, R.S.; Lin, C.C.; Tsao, H.Y. Empowering Large Language Models to Leverage Domain-Specific Knowledge in E-Learning. *Appl. Sci.* **2024**, *14*, 5264. [CrossRef]
65. Radeva, I.; Popchev, I.; Doukovska, L.; Dimitrova, M. Web Application for Retrieval-Augmented Generation: Implementation and Testing. *Electronics* **2024**, *13*, 1361. [CrossRef]
66. Lewis, P.; Perez, E.; Piktus, A.; Petroni, F.; Karpukhin, V.; Goyal, N.; Küttler, H.; Lewis, M.; Yih, W.T.; Rocktäschel, T.; et al. Retrieval-Augmented Generation for Knowledge-Intensive NLP Tasks. In Proceedings of the 34th International Conference on Neural Information Processing Systems, New York, NY, USA, 6–12 December 2020; pp. 9459–9474.
67. Li, R.; Liu, M.; Xu, D.; Gao, J.; Wu, F.; Zhu, L. *A Review of Machine Learning Algorithms for Text Classification. Cyber Security*; Lu, W., Zhang, Y., Wen, W., Yan, H., Li, C., Eds.; Springer Nature: Singapore, 2022; pp. 226–234.
68. Munir, K.; Sheraz Anjum, M. The Use of Ontologies for Effective Knowledge Modelling and Information Retrieval. *Appl. Comput. Inform.* **2018**, *14*, 116–126. [CrossRef]
69. OpenAI GPT 3.5 Turbo. Available online: https://platform.openai.com/docs/models/gpt-3-5-turbo (accessed on 28 June 2024).
70. OpenAI GPT-4o. Available online: https://platform.openai.com/docs/models/gpt-4o (accessed on 22 July 2024).

71. Sharma, A. 11 Best Generative AI Tools and Platforms. Available online: https://www.turing.com/resources/generative-ai-tools (accessed on 6 July 2024).
72. Kothari, S. Top Generative AI Tools: Boost Your Creativity. Available online: https://www.simplilearn.com/tutorials/artificial-intelligence-tutorial/top-generative-ai-tools (accessed on 6 July 2024).
73. Techvify Team GPT-3.5 vs. GPT-4: Exploring Unique AI Capabilities. Available online: https://techvify-software.com/gpt-3-5-vs-gpt-4/ (accessed on 14 July 2024).
74. Prakoso, D.W.; Abdi, A.; Amrit, C. Short Text Similarity Measurement Methods: A Review. *Soft Comput.* **2021**, *25*, 4699–4723. [CrossRef]
75. Kaggle Kaggle Datasets. Available online: https://www.kaggle.com/datasets (accessed on 25 July 2024).
76. GENSIM. Available online: https://radimrehurek.com/gensim/models/word2vec.html (accessed on 25 July 2024).
77. Dal Pont, T.R.; Sabo, I.C.; Hübner, J.F.; Rover, A.J. Impact of Text Specificity and Size on Word Embeddings Performance: An Empirical Evaluation in Brazilian Legal Domain. In Proceedings of the Intelligent Systems: 9th Brazilian Conference, BRACIS 2020, Rio Grande, Brazil, 20–23 October 2020; Proceedings, Part I. Springer Verlag: Berlin/Heidelberg, Germany, 2020; pp. 521–535.

Disclaimer/Publisher's Note: The statements, opinions and data contained in all publications are solely those of the individual author(s) and contributor(s) and not of MDPI and/or the editor(s). MDPI and/or the editor(s) disclaim responsibility for any injury to people or property resulting from any ideas, methods, instructions or products referred to in the content.