CSF 407 – ARTIFICIAL INTELLIGENCE

ENHANCING TEXT-PATTERN SEARCH IN SQL DATABASES

USING

TF-IDF, VSM, BM25, Trie, Boolean Search

TEAM MEMBERS ID	NAMES
1 2021A7PS2682H	ANIRUDH BAGALKOTKER
2 2021A3PS3203H	SAKAR HIRDE
3	
4	

1. INTRODUCTION

INTRODUCTION TO TEXT SEARCHING IN SQL

Searching for text patterns in Relational Databases is a common task, especially in SQL Databases like MYSQL, POSTGRESQL, SQLLITE which are commerically and widely used for various applications like webistes, search engines. Data mining and information retrieval. In the ever-evolving landscape of data management and retrieval, the efficient searching of text patterns within SQL Databases has become a critical aspect.

SQL (Structured Query Language) Databases are widely used for storing and managing structured data, but when it comes to texts and the traditional search algorithm implemented in SQL using methods like “MATCH” SQL Query, they often fall short in terms of accuracy.

In this Project, will try to address the challenges assosiated with the Text and Pattern Search in SQL Databases by finding alternative Search Algorithms which can improve the Efficiency and the Quality of the Solution.

THE PROBLEM

The Primary problem which we are looking to address is the sub-optimal performance of the in-built SQL Queries which are generally used for text pattern search. While SQL Databases are exceptionally powerful for structured dta retrieval, they have limitations when it comes to seacrhing text-patterns. The Traditional SQL Queries often struggle to efficiently locate and retrieve relavent information form long text fields, especially when dealing with extensive databases which vontain a vast amount of textual content, such as documents, articles, logs or user-generated content. The most popular SQL Query used for text-pattern searching is “MATCH” but lets look into all of the built-in methods available in SQL which can be used for text-pattern searching.

1. LIKE: The Like operator can be used to search strings that match a specific pattern. The pattern can be constructed using wildcards, such as the percent(%) and underscore(_) charecters.

2. SIMILAR TO: The SIMILAR TO operator is similar to the LIKE operators, but compared to LIKE, it takes into account the similarity of the strings being compared. This can be usefull for finding text that are similar in spelling or pronounciation.

3. CONTAINS: The CONTAINS function is a full-text search function that can be used to search function that can be used to search for strings that contain a specific word or phrase.

The CONTAINS function is more efficient than MATCH when searching long texts, but CONTAINS cant find a pattern in text, i.e. It can only find the whole exact word.

4. MATCH: The MATCH function uses a full-text search algorithm to search for patterns in text. The algorithm takes into account the context of the words in search patterns. While it could sound the best of all, it is quite slow and inefficient.

As we can see for the text-pattern search, the ideal available method is MATCH, but now let us discuss some problems associated with it.

1. **LIMITED FLEXIBILTY:** SQL's native "MATCH" Query is often designed for simple text pattern matching, lacking the flexibility to handle complex and varied search requirements.

2. **PERFORMANCE BOTTLENECKS:** As the size of the dataset grows, the performance of text searches using "MATCH" can degrade significantly. This becomes a substantial challenge when dealing with databases that contain lots of records.

3. **LACK OF RANKING:** There are no robust ranking capabilities, making it difficult to prioritise and retrieve the most relevant results when searching for text patterns.

4. **SCALABILITY:** Today's world is data-driven, and what matters to us the most is the scalability. Scalability of text search operations is quite crucial, and the Traditional SQL text search approaches may struggle to keep the demands of large-search, rapidly growing databases.

THE SOLUTION

Let us see some relevent solutions for text-pattern searching:

1. **Full-Text Search:** This core search method enables quick and effective text-based searches. Full-text search functionality is frequently built into SQL databases (for example, MATCH...AGAINST in MySQL, CONTAINS in SQL Server), which we already saw before and discussed to find another solution.

2. **Fuzzy Search:** Fuzzy search techniques assist in locating outcomes that roughly match the search query. Soundex and Levenshtein distance are two common methods.

3. **Vector Space Model (VSM):** VSM is a useful information tool that represents documents and queries as vectors in multi-dimensional space.

4. **Inverse Document Frequency Term Frequency (TF-IDF):** TF-IDF is a statistical tool for assessing a word's significance in relation to a group of documents. It is frequently applied to text-based searches.

5. **BM25:** BM25 is an enhanced version of TF-IDF that takes into account things like term saturation and normalisation of document length. It is well known for being efficient at retrieving information.

6. **Prefix Trees (Trie):** Trie structures can be utilised for prefix-based searches and autocomplete. For both storing and recovering words or phrases, they are effective.

7. **Clustering Algorithms:** Search results can be organised into useful groups using clustering techniques like k-means or hierarchical clustering.

8. **Boolean Search:** Boolean search combines keywords using operators like AND, OR, and NOT to produce more relevant results. It is often used for precise searching.

2. LITERATURE SURVEY

This section presents an overview of some recent research activities on text pattern searching algorithms. We would first discuss the search algorithms that are already implemented for SQL Methods like “MATCH”.

In ‘Boyer-Moore approach to approximate string matching’, the article thoroughly assesses the literature on compressed pattern matching, with an emphasis on using the Boyer-Moore approach in the context of college systems. The Boyer-Moore algorithm is directly used in the “MATCH” SQL Query algorithm. It emphasizes two specific objectives: Goal 1, which aims to search in compressed files more quickly than conventional decompression and searching, and Goal 2, which aims to make compressed data a viable choice for searching more quickly than the original uncompressed files. The paper points out that accomplishing Goal 2 is more difficult since there is a realistic linear relationship between the lengths of compressed and original text, which frequently impairs the efficiency of complex algorithms made for compressed text. The research offers the idea of collage systems, a formal framework that includes multiple dictionary-based compression techniques, to address this issue. Within this context, it presents a generic compressed pattern-matching algorithm and illustrates how the existence of truncation in collage systems may have a major impact on pattern-matching effectiveness. The byte pair encoding (BPE) compression method, which has been comparatively neglected due to sluggish compression and poor compression ratios, is also included in the literature review. Although the paper proposes a method that surpasses more established O(N) time techniques, such as the KMP algorithm and the Shift-Or algorithm, in terms of pattern matching speed, BPE is ideally suited for accelerating pattern matching. The study summarizes the body of work in the field of compressed pattern matching, highlighting the significance of taking into account cutting-edge methods like collage systems and BPE to speed up pattern matching in compressed text while defying conventional compression standards.

"A Boyer—Moore Type Algorithm for Compressed Pattern Matching", a conference paper, discusses The Boyer-Moore technique, which is well-known for its effectiveness in precise string matching, is presented in this article as a novel extension to the area of approximation string matching. The "k mismatches" problem, where the objective is to find instances of a pattern in a text with at most k character mismatches, and the "k differences" problem, which expands the concept to include insertions, deletions, and changes within the permitted k differences, are two variations of the problem that are investigated. The Boyer-Moore method is sublinear, and the study emphasizes how efficiency increases with more extended patterns and bigger alphabets. For both issues, a number of techniques are examined, including enhancements for random strings and dynamic programming solutions with temporal complexity of O(mn). For both the k differences problem and the k mismatches problem, the suggested Boyer-Moore-based method is presented. The study estimates the temporal complexity of these algorithms,

taking into consideration the length of the pattern, the number of permitted deviations, and the size of the alphabet. In particular, for big alphabets and pattern lengths, experimental comparisons show that Boyer-Moore-based algorithms perform better. Overall, this study increases the Boyer-Moore algorithm's efficiency for approximation string matching, offering excellent answers for real-world problems. The work is divided into parts that cover the k mismatches problem, the k differences problem, and experimental findings, providing in-depth analyses of the effectiveness of the suggested algorithms.

The study "Adapting the Knuth–Morris–Pratt algorithm for pattern matching in Huffman encoded texts" tackles the difficulty of compressing pattern matching in binary Huffman-encoded texts. The goal is to find patterns in compressed texts without completely decompressing them. The authors suggest a customized version of the Knuth-Morris-Pratt (KMP) method for this problem. The Huffman-encoded alphabet is binary. Thus, the modified KMP method uses this by processing one extra bit in case of a mismatch. A preprocessed table ensures that pattern matching always lines up with the beginning of codewords, preventing false matches brought on by codeword borders. The effectiveness of compressed pattern matching is also discussed as it relates to integrating the modified KMP algorithm with two practical Huffman decoding schemes: skeleton trees and numerical comparisons. Experiments show that these algorithms outperform the conventional "decompress then search" strategy, enabling the retention of compressed data while conserving memory. The literature review in this study briefly discusses several related topics, such as word-based Huffman compression methods, probabilistic algorithms for searching Huffman encoded texts, and Aho-Corasick pattern matching. Additionally, it outlines the benefits and drawbacks of each strategy and offers experimental contrasts for assessing performance. In conclusion, this study thoroughly analyses the state-of-the-art in Huffman encoded texts compressed pattern matching, giving workable solutions and proving their viability through testing.

Now, let’s see few text-pattern recognising/searching research papers that use different algorithms to observe and search the patterns in text and then retrieving the result in a efficient manner. These algorithms are currently not adopted by SQL for any of the methods that involve searching of texts or patterns.

The study “The problem of fuzzy duplicate detection of large texts” discusses the subject of fuzzy duplication detection in textual data and investigates different methods to solve this problem. These methods are divided into three groups: fuzzy search algorithms without data indexing, fuzzy search algorithms with data indexing, and distance measurements between strings. The study provides an in-depth analysis of current approaches for fuzzy duplication detection, highlighting their advantages and disadvantages. The paper's main contribution is introducing a method for fuzzy duplication detection. Within the AVTOR.NET system, this algorithm is used. It uses several text preparation techniques, such as character set conversion, text encoding into single lengthy strings, and other cleaning operations, including removing HTML tags, punctuation, and cases. Furthermore, character substitution and stemming are used to increase the algorithm's efficiency. The idea of shingles serves as the foundation for the fuzzy

duplication detection technique presented in the study. The text is broken up into word sequences, and MD5 codes (shingles) are computed for each sequence. The amount of matched MD5 codes indicates how similar the two documents are to one another. The documents are regarded as complete duplicates if all codes match. No results imply entirely different papers, whereas partial matches suggest fuzzy duplication. The report also shows how the AVTOR.NET technology is used in real-world situations to assess graduation projects for plagiarism. Significant document handling efficiency is shown for the system, highlighting its usefulness in practical situations. In conclusion, the paper presents a thorough literature review of fuzzy duplicate detection techniques, introduces an algorithm used in the AVTOR.NET system, and emphasizes its real-world use in detecting plagiarism, constituting a significant contribution to the fields of text analysis and information retrieval.

The study “Automatic classification of Tamil documents using vector space model and artificial neural network ” investigates how the morphologically diverse Dravidian classical language of Tamil might be classified using the Vector Space Model (VSM) and Artificial Neural Networks (ANN). The requirement for automatic classification of such papers has grown more crucial as a result of the internet's exponential rise of electronic documents in regional languages like Tamil. For model training and testing, the study makes use of a corpus made available by the Central Institute of Indian Languages (CIIL). The Term Frequency-Inverse Document Frequency (TFIDF) weighting method is used in the VSM methodology, where documents are represented as high-dimensional vectors and phrases are given weights depending on their significance. In contrast, a neural network with hyperbolic tangent activation in the hidden layer is utilised for classification in the ANN model. According to experimental data, the ANN model surpasses the VSM model, classifying Tamil documents with an accuracy rate of 93.33% as opposed to 90.33% for the VSM model. The study emphasises the capability of neural networks to capture non-linear relationships within the document vectors and the possibility of expanding these models for more extensive document collections. By shedding light on efficient techniques for automatically categorising texts in a language with complex morphological features, such as Tamil, this research contributes to the field of text classification.

The study “Automatic summarization of YouTube video transcription text using term requency-inverse document frequency” is all about automatic summary, which is helpful in reducing lengthy texts or videos to more manageable lengths. This method is crucial for assisting people in swiftly understanding the most critical information from long publications or movies. This study aimed to automate text summarization using natural language processing techniques, especially for YouTube videos. In order to extract crucial keywords for summarization, the term frequency-inverse document frequency (TF-IDF) method was used after transcribing the films. The main goal was to cater to students and researchers who do not have the time to watch complete films by making long movies more accessible by offering succinct text summaries. The Rouge approach, which assesses the calibre of the produced summaries, was used in the study. The convolutional neural network (CNN)-dailymail-master dataset was used for testing and evaluation. Regarding the bigger picture, the abundance of internet material has made automatic text summaries

more critical. A practical document summary is necessary, according to researchers Ajmal and Haroon. In this context, a summary is a much-condensed book version that attempts to convey the key concepts. The difficulty in automatically constructing text summaries is in maintaining the main ideas and information. Several strategies, including extractive and abstractive approaches, have been developed to address this issue. These techniques use algorithms to create summaries by examining the text's structure, keywords, and semantic linkages. The study also emphasized earlier efforts at automated text summary, including natural language processing, latent semantic analysis, and rhetorical structure theory for summarizing Arabic text. The study also highlighted the significance of video summarization, which entails movie transcription and using computer methods to examine the material. Researchers can learn more about video content by analyzing word connections and commonly used terms or phrases in video transcripts. In conclusion, this study proposed an automated text summarization technique focused on TF-IDF analysis and YouTube video transcriptions. It highlighted the value of automatic summarizing in the age of voluminous internet material and assessed the efficiency of the strategy using the Rouge technique.

The paper “Text Embeddings by Weakly-Supervised Contrastive Pre-training” describes the text embedding model E5, trained using contrastive learning on the large text pair dataset CCPairs, which is presented in the study. E5 excels in retrieval, clustering, and classification problems and offers flexible single-vector representations for texts. On BEIR benchmarks, it exceeds the BM25 baseline in zero-shot retrieval, and under fine-tuned settings, it achieves cutting-edge performance on the MTEB benchmark. Contributions made by E5 include a revolutionary consistency-based data filtering algorithm and its approach to high-quality data curation from a variety of sources, including CommunityQA, Common Crawl, and Scientific publications. The detailed analysis of related literature included in the research emphasizes the value of high-quality data as well as the efficiency of self-supervised pre-training for text embeddings. Compared to bigger models like GPT-3 and BERT, E5 achieves comparable results even with smaller model sizes, highlighting the importance that E5 has played in improving the field of text embeddings.

The study “Comparison of the BM25 and rabinkarp algorithm for plagiarism detection” examines plagiarism detection techniques using the BM25 and Rabin Karp algorithms as its main point of comparison. In academics, plagiarism is a major problem, and effective detection is essential. The efficacy of BM25, a term weighting method, and Rabin Karp, a hashing-based strategy, is examined. In order to prepare text data for analysis, the study underlines the value of preprocessing techniques such as folding, cleaning, tokenizing, filtering, and stemming. The findings suggest that whereas Rabin Karp often produces greater similarity values, BM25 tends to provide similarity values that are closer to the actual word ratio in test documents. However, because of its effective hashing method, Rabin Karp beats BM25 in terms of execution time. The article highlights the necessity for more study to evaluate the efficacy of plagiarism detection and makes recommendations for possible Rabin Karp algorithm improvements. Overall, the study offers important new perspectives on plagiarism detection techniques and lays the groundwork for further advancements in this crucial area.

The work on “Dynamic Dictionary Matching” tackles the problem of effectively searching for instances of these patterns in a given text while managing a developing list of pattern strings within a dictionary. This study presents unique strategies, particularly in the areas of dynamic prefix-trees (Trie) management and pattern splitting, by drawing inspiration from fundamental research in the field of string processing algorithms and data structures. These techniques are used to build and update suffix trees that accurately depict the changing vocabulary, enabling insertion, deletion, and search operations with temporal complexities that are more sophisticated than those of current algorithms. By combining these approaches into a single system that can handle dynamic dictionary changes while also conducting effective pattern searches inside textual material, this work considerably improves upon earlier studies. The method has wide-ranging potential applications, including but not limited to molecular biology research and scanning bibliographic databases. This literature overview highlights the research's larger implications for string processing algorithms and creates new opportunities for investigation and practical use.

The TFIDF feature selection approach, “Improved feature selection approach TFIDF in text mining”, used in the context of text mining, is presented in this study. The objective is to develop a Vector Space Model (VSM) that, by effectively modelling text data, aids text classification. The classification findings are used to assess the precision of this TFIDF-based technique, highlighting its advantages and disadvantages. The research offers an improved TFIDF-based feature selection method to increase accuracy in response to empirical data. Text mining requires converting textual data into structured feature vectors since it differs from typical data mining in that it is unstructured or only partially structured. The use of the term weight as a feature selection criterion is specifically covered in the paper's discussion of data preparation approaches. With modifications to improve its performance, the TFIDF approach is used for feature selection. The study emphasizes the significance of feature subset selection and the efficiency of the TFIDF technique when used in conjunction with Vector Space Modeling to handle text data for classification. The method's effectiveness is demonstrated by experimental findings, which achieve a classification accuracy of 76% and further improvements when feature weights are adjusted with mutual information. The results highlight the importance of data pretreatment in text mining and offer directions for further study to improve classification accuracy and effectiveness.

The study “Subjectivity Classification of Filipino Text with Features based Frequency – Inverse Document Frequency” discusses a machine learning method for classifying subjectivity in Filipino texts, both at the document and sentence levels. Its main goal is to identify which phrases in a text are subjective and if that document contains any subjective information. The Philippine Star provided the authors with a sizable dataset of news and opinion pieces, and they utilised the Term Frequency-Inverse Document Frequency (TF-IDF) scores to choose the most pertinent terms as features. They used machine learning methods, including C4.5, Naive Bayes, k-Nearest Neighbor, and Support Vector Machine (SVM) for categorisation. High accuracy was attained while classifying subjectivity at the document level, with SVM performing best (95.06%). On the other hand, the subjectivity categorisation outcomes at the phrase level varied less between algorithms.

The research discovered that while the selected deciding words were successful at classifying documents, they were less successful at classifying sentences. The study emphasises the significance of feature selection and the effects of several machine learning algorithms on the categorisation of subjectivity in Filipino literature. Future research directions are suggested, including TF-IDF at the phrase level, testing models across domains, and experimenting with thresholds.

The study “A Survey on Text Mining- techniques and application” explores the complex domain of text mining, a methodology developed to extract important insights, information, and patterns from unstructured textual data derived from many sources. The focus is mostly on the financial sector, which offers a substantial amount of data pertaining to the financial performance of organizations. Although the automated analysis of financial statistics is widely practised, the task of extracting meaningful insights from the textual components of financial reports has remained a persistent and ongoing difficulty. The narrative component of an annual report frequently contains more comprehensive information compared to conventional financial figures. In order to tackle this difficulty, the study integrates data mining techniques to analyze a combination of quantitative and qualitative data derived from financial reports. The methodology utilizes self-organizing maps for quantitative analysis and prototype-matching text clustering for qualitative analysis. The study centres on the examination of quarterly reports derived from three important telecommunications businesses within the industry. The primary methodologies examined are classification, clustering, information extraction, and information visualization, acknowledging the significance of natural language processing (NLP) and topic tracking in the context of text mining procedures. Text mining has a wide range of applications that span several disciplines, such as business intelligence, bioinformatics, security, human resource management, online search improvement, publishing, telecommunications, healthcare, and finance. In summary, this study highlights the importance of text mining as a potent instrument for extracting practical insights from unstructured textual data. This statement underscores the capacity of text mining to fundamentally transform decision-making processes and the exploration of information across many disciplines, eventually facilitating well-informed decision-making and the advancement of data-driven initiatives.

The present research, “A graph-based multi-level linguistic representation for document understanding”, provides a thorough examination of the assessment of Question Answering for Machine Reading Comprehension (QA4MRE) within the framework of two separate datasets originating from the years 2011 and 2012. The datasets encompass a diverse array of subjects, such as Climate Change, Music and society, Alzheimer's disease, and AIDS, each consisting of 10 inquiries per topic. The present study assesses many methodologies, namely MinText_TreeTagger, MinText_Lancaster, MinText_Synonym, MinText_Hyponym, and Without_MinText, throughout two consecutive years. The findings demonstrate the inherent difficulties associated with the QA4MRE work, as seen by the variable levels of performance observed across different methodologies. The limits of the technique are emphasized, particularly in situations when wrong responses are chosen as a result of simplistic node selection and a lack of information regarding path

length. The paper highlights the necessity of employing more advanced methodologies and algorithms in order to augment precision. Regarding future prospects, the paper proposes the use of the graph-based multi-level linguistic representation for additional natural language processing (NLP) tasks, including textual entailment and semantic similarity. Furthermore, this highlights the significance of including supplementary semantic links in the representation and assessing various configurations of parsing tools. Additionally, the research examines the possible incorporation of text tagging, specifically named entity recognition, inside the proposed framework. In general, this work offers significant insights into the intricacies of the QA4MRE job and establishes a foundation for future investigations in the fields of natural language processing and machine reading comprehension.

The present research, “A comparative study of TFIDF, LSI and multi-words for text classification” explores the crucial domain of text representation within the context of text mining, which is a vital aspect of natural language processing. This study examines three well-recognized techniques for text representation: Term Frequency-Inverse Document Frequency (TFIDF), Latent Semantic Indexing (LSI), and multi-word representation. The TFIDF approach, well recognized in academic literature, calculates term weights by considering the frequency of occurrence and the overall significance of terms within a corpus. In contrast, Latent Semantic Indexing (LSI) aims to reveal underlying semantic patterns by utilizing singular value decomposition, demonstrating enhanced efficacy in certain undertakings. The concept of multi-word representation encompasses examining multi-word expressions, which involves considering both semantic and statistical aspects to improve text representation. The effectiveness of these approaches is heavily emphasized in a wide range of literature, highlighting their crucial significance in information retrieval and text classification. Scholars have also investigated hybrid ways that effectively integrate various strategies. In conclusion, text representation plays a pivotal role in determining the effectiveness of text mining applications. The choice of technique for text representation depends on the unique characteristics of the data, the tasks at hand, and the objectives to be achieved.

Significant research, “Text Mining in Radiology Reports” has recently been put into radiology report analysis and medical text mining. The Medical Language Extraction and Encoding System (MedLEE), created by Friedman et al., is a noteworthy technology that structures radiology reports using semantic approaches and lexicons to achieve high accuracy and recall in encoding medical information. Taira et al. focused mainly on dependency diagram construction in their investigation of statistical parsing methods for obtaining medical results from reports. Another system, NeuRadIR, was developed by Dominic et al. for neuroradiological information retrieval. It offered a variety of retrieval modalities but had vocabulary coverage issues. Also becoming more well-liked are content-based image retrieval (CBIR) systems for medical pictures. Despite being independent procedures from text analysis, Sinha et al. and Lacoste et al. used medical terminology and free-text reports from the Unified Medical Language System (UMLS) for picture indexing. Together, these studies highlight the value of structured radiology reports, the necessity of

NLP, and the potential advantages of combining text and picture data in medical data mining and retrieval systems.

The study “A Text Search System Using Boolean Strategies for the Identification of Infrared Spectra” focuses on the text search method for a minicomputer-based system is developed and used in this paper, emphasizing the analysis of chemical condensates. This flexible technique can be used for many datasets with the right preprocessing. The article shows how to search a sizable collection of 91,875 infrared spectra in ASTM format using standard text searching methods, such as truncation and Boolean logic. This text-based searching technique permits adding more data to the search input, such as compound names, molecular formulae, and information on chemical functionality. By enabling the selection of specific subsets of spectra for additional study using pattern recognition or statistical methods, the system expands its value beyond conventional library searches. The study of the literature provides a background on the development of text-based search systems, stressing the shift from card sorting to computerized strategies for effective data retrieval. It draws attention to the difficulties involved in examining spectral data, particularly when navigating the subtleties of peak intensities and forms. The suggested text-oriented search system is discussed in the study, along with its advantages for handling chemical data, and its independence from a particular data source makes it adaptable for various literature databases. Last, but not least, it highlights the algorithm's value in improving the precision and selectivity of spectrum searches while providing flexibility for multiple applications outside of typical library matching.

3. CONCLUSION

This extensive literature review of the AI Project has helped us explore several academic endeavors and investigations pertaining to algorithms for finding and recognizing patterns in text. Numerous research have investigated diverse strategies and methodologies aimed at enhancing text and pattern searching capabilities, encompassing SQL methods such as the "MATCH" function as well as more comprehensive contexts.

The findings derived from the scholarly investigation on SQL techniques such as "MATCH" have provided insights into the difficulties included in conducting text pattern searches within SQL databases, as well as the potential remedies that have been suggested to tackle these difficulties. The methods employed to improve the efficiency and accuracy of text pattern matching encompass Boyer-Moore, Knuth-Morris-Pratt, and Rabin-Karp.

Furthermore, we have conducted investigations that extend beyond the use of SQL procedures and instead incorporate other algorithms and strategies for the purpose of text pattern identification and searching. The aforementioned research encompass a broad range of applications, spanning from fuzzy duplication detection to document categorization, summarization, subjective classification, and several others. Each of the aforementioned research provides useful insights into the intricacies and subtleties of text mining, natural language processing, and machine learning.

The study reveals that the subject of text pattern finding and recognition is undergoing constant evolution, as academics are actively devising novel algorithms and approaches to tackle diverse issues across numerous fields. The significance of feature selection, data preparation, and algorithm selection in attaining precise and efficient text pattern recognition has also been noted.

In the future, our intention is to apply and evaluate various strategies and algorithms that have been examined in these research in order to improve the efficiency of text pattern searching in SQL databases and other pertinent applications. The proposed implementation would adopt a sequential method, incorporating a visual representation in the form of a graphical timeline, which is outlined as follows:

REFERENCES / BIBLIOGRAPHY

Agrawal, D., El Abbadi, A., Salem, K., Bravo, M., Diegues, N., Zeng, J., ... & Ahmad, Y. (1995). Data engineering.

Tarhio, J., & Ukkonen, E. (1990, January 1). *Boyer-Moore approach to approximate string matching*. SpringerLink. https://link.springer.com/chapter/10.1007/3-540-52846-6_103

Shibata, Y., Matsumoto, T., Takeda, M., Shinohara, A., & Arikawa, S. (1970, January 1). *A Boyer-Moore type algorithm for compressed pattern matching*. SpringerLink. https://link.springer.com/chapter/10.1007/3-540-45123-4_17

Pergamon. (2005, May 10). *Adapting the knuth–morris–Pratt algorithm for pattern matching in Huffman encoded texts*. Information Processing & Management. https://www.sciencedirect.com/science/article/pii/S0306457305000191?casa_token=K_i3tA-phbkAAAAA%3Ak2BtTmAh_7rNqbuypzfz7KuVOjHqz8iTIQksfU8ra22ZhT2Y0E7l5-G2bqpq8_c5PDfO_piJeg

Sharapova, E. V., & Sharapov, R. V. (1970, January 1). *The problem of fuzzy duplicate detection of large texts*. The problem of fuzzy duplicate detection of large texts E.V. Sharapova, R.V. Sharapov,. http://repo.ssau.ru/handle/Informacionnye-tehnologii-i-nanotehnologii/The-problem-of-fuzzy-duplicate-detection-of-large-texts-69667

Salton, G., Tan, S., Baum, E. B., Fabrizio, S., & Joachims, T. (2009, February 16). *Automatic classification of Tamil documents using vector space model and Artificial Neural Network*. Expert Systems with Applications. https://www.sciencedirect.com/science/article/abs/pii/S0957417409001547?casa_token=

O6v2me5paXEAAAAA%3A6HiWoVVCAUH4ymJ2GAiJtLQ0_nsKp1wJl0Q9n1loObBaprRRFpOk8cPp3o8TdGRZuL0MzFqLFw

Albeer, R. A., Al-Shahad, H. F., Aleqabie, H. J., & Al-shakarchy, N. D. (2022). Automatic summarization of YouTube video transcription text using term frequency-inverse document frequency. Indonesian Journal of Electrical Engineering and Computer Science, 26(3), 1512-1519.

Wang, L., Yang, N., Huang, X., Jiao, B., Yang, L., Jiang, D., Majumder, R., & Wei, F. (2022, December 7). *Text embeddings by weakly-supervised contrastive pre-training*. arXiv.org. https://arxiv.org/abs/2212.03533

Wijaya, I. N. S. W., Seputra, K. A., & Parwita, W. G. S. (2021, March). Comparison of the BM25 and rabinkarp algorithm for plagiarism detection. In Journal of Physics: Conference Series (Vol. 1810, No. 1, p. 012032). IOP Publishing.

Amir, A., Farach, M., Galil, Z., Giancarlo, R., & Park, K. (1994). Dynamic dictionary matching. Journal of Computer and System Sciences, 49(2), 208-222.

Li-Ping Jing, Hou-Kuan Huang and Hong-Bo Shi, "Improved feature selection approach TFIDF in text mining," Proceedings. International Conference on Machine Learning and Cybernetics, Beijing, China, 2002, pp. 944-946 vol.2, doi: 10.1109/ICMLC.2002.1174522.

R. V. J. Regalado, J. L. Chua, J. L. Co and T. J. Z. Tiam-Lee, "Subjectivity Classification of Filipino Text with Features Based on Term Frequency -- Inverse Document Frequency," 2013 International Conference on Asian Language Processing, Urumqi, China, 2013, pp. 113-116, doi: 10.1109/IALP.2013.40.

Inzalkar, S., & Sharma, J. (2015). A survey on text mining-techniques and application. International Journal of Research In Science & Engineering, 24, 1-14.

Pinto, D., Gómez-Adorno, H., Vilarino, D., & Singh, V. K. (2014). A graph-based multi-level linguistic representation for document understanding. Pattern recognition letters, 41, 93-102.

T. Gong et al., "Text Mining in Radiology Reports," 2008 Eighth IEEE International Conference on Data Mining, Pisa, Italy, 2008, pp. 815-820, doi: 10.1109/ICDM.2008.150.

Woodruff, H. B., Lowry, S. R., & Isenhour, T. L. (1975). A Text Search System Using Boolean Strategies for the Identification of Infrared Spectra. Journal of Chemical Information and Computer Sciences, 15(4), 207-212.