Mathematical model of a text document with consideration of time for search algorithm

Iryna Yurchuk[1,†], Kseniia Dukhnovska[1,*,†], Oksana Kovtun[1,†], Anna Martsafei[1,†] and Anastasiia Kushnir[1,†]

[1] *Taras Shevchenko National University of Kyiv, 60 Volodymyrska Street, Kyiv, 01033, Ukraine*

Abstract

This research examines mathematical models for text documents, incorporating temporal factors to improve the efficacy of search algorithms. In light of the exponential growth of information in both digital and physical realms, it is imperative to acknowledge the dynamic nature of information. A dynamic model based on TF-IDF measurements has been developed and experimentally validated within a Python 3.8 environment using the Spider framework. Results demonstrate that this dynamic model can significantly enhance the precision of text document search.

Keywords

TF-IDF, search engine algorithms, text document model

1. Introduction

The exponential proliferation of textual data in the contemporary digital age necessitates the creation of novel, highly efficient information retrieval algorithms. To guarantee the precision and pertinence of search outcomes, it is imperative to accurately model not only the syntactic composition of texts but also their semantic meaning. This research focuses on a mathematical model that incorporates the temporal dimension for text information retrieval algorithms.

2. Literature review and problem statement

Mathematical models underpin modern natural language processing systems. These models enable the transformation of text into digital representations, allowing computers to perform various analytical and generative tasks. Among the most common models are the bag-of-words model, n-grams, vector spaces, and neural networks. The bag-of-words model represents a document as a collection of words without considering their order. Each word corresponds to a dimension in a vector space, and the value of the dimension indicates the frequency of the word in the document. While this model is effective for many tasks, it does not account for syntax or semantics, which can be a limitation for certain applications. The n-gram model is a generalization of the bag-of-words model, considering sequences of n words (n-grams). This allows for capturing some syntactic features of the text. In the vector space model, unlike the bag-of-words model, words are strictly ordered. The coordinates of the vector can be calculated using various methods (TF-IDF, word embeddings). This model enables measuring semantic similarity between words and documents. The most powerful models capable of learning from large amounts of data are neural networks. The

Information Technology and Implementation (IT&I-2024), November 20-21, 2024, Kyiv, Ukraine

* Corresponding author.

† These authors contributed equally.

i.a.yurchuk@gmail.com (I. Yurchuk); duchnov@ukr.net (K. Dukhnovska); kovok@ukr.net (O. Kovtun); annamartsafei@knu.ua (A. Martsafei); anastasiia_kushnir@knu.ua (A. Kushnir)

0000-0001-8206-3395 (I. Yurchuk); 0000-0002-4539-159X (K. Dukhnovska); 0000-0003-0871-5097 (O. Kovtun); 0009-0003-1943-9048 (A. Martsafei); 0009-0002-7640-7344 (A. Kushnir)

primary types of neural networks used for text processing are recurrent neural networks (RNNs), long short-term memory (LSTM) networks, gated recurrent units (GRUs), transformers, and convolutional neural networks (CNNs).

The selection of an appropriate model is contingent upon the specific task at hand and the desired level of precision. For instance, a study [1] proposes a mathematical text model utilizing hesitant fuzzy sets (HFS). Deviating from vector space models where each word is considered as a dimension, this approach clusters words into fuzzy sets of synonyms. The model is employed to retrieve of mathematical information, which is primarily conveyed through mathematical formulas. The paper posits that mathematical formulas serve as a "semi-formal visual language". The authors subsequently apply algorithms for local semantic distillation, grounded in the "teacher-student" framework, to facilitate the search for texts containing mathematical formulas.

In [2] by the author, a practical application is investigated. Web scraping is employed to gather data. Subsequently, named entities are extracted from the resulting textual data using SD-NER algorithms. The semantic disambiguation model for named entity recognition leverages semantic knowledge to enhance the precision of named entity identification within text by considering contextual and semantic cues. The statistical disambiguation model for named entity recognition utilizes statistical methods to train on extensive datasets and determine the most likely tags for named entities. The proposed SD-NER model attained an F_1-score of 0.920 on the binary classification task.

Research [3] investigates the challenge of storing vast quantities of digital information. The research compares the performance of various clustering algorithms and mathematical models within the context of information retrieval for textual documents. In particular, the study contrasts centroid-based clustering with clustering augmented by Latent Semantic Analysis (LSA). The hypothesis is that LSA enhances clustering by reducing the semantic distance between related terms in the vector space. The researchers conclude that clustering outcomes are contingent upon the document representation and the similarity metric employed. When applied to short documents, LSA does not yield superior results compared to conventional clustering techniques.

The authors [4] focused on the clustering of web news articles related to the latest technologies. The study proposes a novel document representation model, Bag-of-Near-Synonyms (BoNS), which is based on the concept of creating clusters of lexical units with similar meanings (quasi-synonyms) by applying word embeddings and agglomerative clustering algorithms. Subsequently, a vector space model, Set Frequency-Inverse Document Frequency (SF-IDF), is employed to represent documents, where each coordinate corresponds not to an individual word but to a cluster of quasi-synonyms. To accelerate computations, a hashed modification of SF-IDF (hSF-IDF) is proposed, which allows for linear calculation of SF values by applying a hash function to map each cluster to a unique identifier. The effectiveness of the proposed model is demonstrated on the task of online clustering of news about Chinese technologies, for which an improved batch algorithm is developed. Experimental results on real-world data indicate the superiority of the proposed model over traditional methods such as TF-IDF, average word or character embeddings, Latent Dirichlet Allocation (LDA), and the "bag of concepts" model, both in terms of accuracy and computational efficiency.

In [5], there is devoted to a comparative analysis of relevant feedback models based on vector space and probabilistic models. The authors of the study consider two main approaches to query modification: term reweighting and query expansion. It is experimentally demonstrated that both approaches can significantly improve search effectiveness. The differences between the considered models lie in the methods of computing term weights and query expansion mechanisms. In particular, the vector space model involves automatic term reweighting based on their frequency in relevant and non-relevant documents, while the probabilistic model employs more complex statistical methods.

Work [6] proposes a novel sentiment analysis model based on a modified TF-IDF algorithm. The authors introduce informativeness weight and word density weight to more accurately determine the emotional connotation of text. The model construction methodology involves the use of a set of

sentiment words as a basis for determining the emotional value of lexical units. Experimental results demonstrate the high effectiveness of the proposed approach, especially in classification tasks and feature word selection.

Text classification is a pressing issue in modern natural language processing. Existing approaches, such as centroid-based classifiers, Naive Bayes, Support Vector Machines, and convolutional neural networks, are constantly being improved. In a study [7], a new method for improving classification accuracy is proposed, which involves combining traditional classifiers with cosine similarity. Experiments conducted on well-known datasets demonstrate a significant improvement in results when using the proposed approach. Additionally, the authors of the study compare the effectiveness of two methods for representing text documents: word counting and TF-IDF.

Research [8] conducts a topological analysis of the sound structure of tongue twisters. The authors apply the mathematical apparatus of topology to identify and formalize complex relationships between sound components of language. The obtained results open up new perspectives for the development of more accurate speech recognition models.

Work [9] proposes a novel approach to text document classification based on an extension of the traditional TF-IDF algorithm using fuzzy logic tools. The introduction of fuzzy logic enables a more accurate modeling of semantic relationships between words and improves classification effectiveness.

The objective of studies [10-11] is to optimize information retrieval by reducing the dimensionality of the search space. To achieve this, the authors propose a new data representation model that involves transitioning from the traditional TF-IDF vector space to a more semantic knowledge space using genetic algorithms.

To date, there has been no research into mathematical models of text documents that incorporate the time dimension.

3. Purpose of the research

The purpose of this article is to present a dynamic model of a collection of documents. To fulfill the objective, a series of tasks were identified and carried out:

- Develop a mathematical model of a text document considering the time factor.
- Investigate a search algorithm for this model.

4. Preliminary processing of a text document

In the realm of information science, key properties of information are relevance, timeliness, completeness, and accuracy. Relevance pertains to the significance of information at the time of its acquisition. Timeliness refers to the applicability of information to a specific problem at the moment it needs to be solved. Information is considered complete if it is sufficient to address a given problem. Accuracy ensures that information accurately reflects the state of an object, process, or phenomenon.

This paper focuses on the textual representation of information. A text document is defined as information presented in textual form on any material carrier. While individual documents are static, collections of documents form dynamic entities that evolve over time. Examples of dynamic document collections include conference proceedings, thematic websites, and social media posts, where documents are organized chronologically.

The exponential growth and dynamic nature of textual information present challenges such as information overload, redundancy, and noise. To effectively analyze large-scale, dynamic content streams generated by internet sources, novel approaches to modeling text documents are required. In this context, natural language text is treated as a sequence of terms. To analyze dynamic document collections, a dynamic model of a text document is necessary.

To construct models of text documents and their collections, the following standard assumptions are commonly employed:

- Document Order Independence: The order of documents within a collection does not influence their semantic or thematic relationships.
- Term Order Independence: The order of terms within a document does not significantly impact its overall meaning.
- Term Frequency Inverse Document Frequency (TF-IDF): Terms that occur frequently across many documents are less informative than rare terms.
- Word Form Normalization: Words with different morphological forms (e.g., plurals, verb conjugations) are treated as equivalent.
- Lemmatization or Stemming: Reducing words to their base or root form.
- Term Extraction and Keyword Identification: Identifying significant terms and phrases.
- Stop Word Removal: Eliminating common, less informative words (e.g., "the," "and," "of").

A text document can be defined as a finite set of words interconnected by lexical, grammatical, and semantic relationships, forming a coherent informational message. A collection of text documents refers to any set of text documents stored locally, on a server, in the cloud, or within a search engine index.

The initial step in processing a text document involves registration within an electronic storage system. Once registered, the document undergoes link extraction, where hyperlinks are identified and queued for subsequent document retrieval. Next, the document is cleaned by removing control characters and other non-textual elements, resulting in a plain text format. The cleaned text is then subjected to parsing, a process that involves tokenization and term extraction. This stage calculates various metrics essential for document analysis and indexing. The document, or its extracted terms, is subsequently indexed, categorized, and stored in a suitable format within the electronic storage system.

A text document is composed of terms, which are syntactically independent morpheme clusters. Unlike word combinations, terms cannot be further subdivided into independent units. The strong internal cohesion of terms distinguishes them from sentence-level structures.

To account for various word forms of a single lexical unit, lemmatization and stemming algorithms are employed.

Lemmatization is the process of reducing the different forms of a word to one single form in accordance with the grammatical forms of a particular language.

The stemming is a process of linguistic normalization, in which use to the removal of derived affixes. To date, there are many different algorithms for stemming. Particularly notable among them are: Porter's Stemmer, KSTEM algorithms, and n-grams. Porter's Stemmer is an algorithm that does not use word vocabulary. It applies a number of rules by which affixes are cut off, basing from the grammar of the language. Porter's Stemmer works fast, but not unmistakably.

The *KSTEM* algorithm is a morphological analyzer whose work is based on the algorithm for replacing the suffix and searching for the base word from which the source word originated.

The n-gram algorithm is based on the postulate: "If word A coincides with word B, taking into account several errors, then with a high degree of probability they will have at least one common substring of length N". Such substrings whose length is n are called n-grams. At the time of parsing, the word is broken down into n-grams, and then the word falls into lists for each of these n-grams. When searching, the query is also divided into n-grams, and each of them is sequentially searched through the list of words containing the given substring. The purpose of this article is to present a dynamic model of a collection of documents. To fulfill the objective, a series of tasks were identified and carried.

To date, there has been no research into mathematical models of text documents that incorporates the time dimension.

5. Text document models

In the simplest case, only the fact of the presence or absence of a term in a document can be considered in a textual analysis. This document model is called binary. A complication of this model is the approach, where for each term not only its presence is indicated, but also some of its "weight". In this case, the weight can be assigned to words, phrases, or to the basics of words.

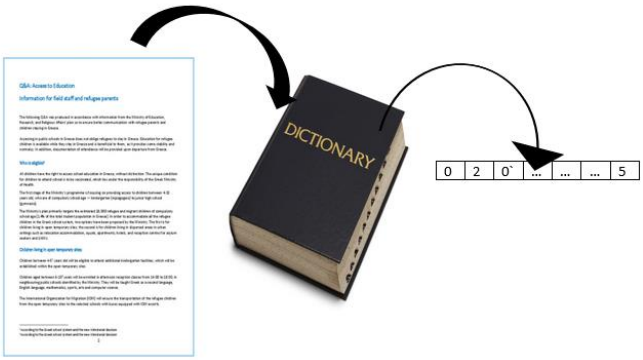

Figure 1: Vector text model.

The simplest method of weighing words in a document is the method of counting the number of occurrences of a term (term weight) in a document (Fig. 1). In this case, it is considered that if a term occurs more frequently in the text of a document, then this document is more likely to be related in content to this term. The disadvantage of this method is that documents of greater length may have a greater weight of the terms included in them. At present, the most common method is to calculate the frequency of the occurrence of terms in a document (TF). Frequency is the ratio of the number of occurrences of a term in the text of a document to the total number of terms of this text. The disadvantage is that here, on the contrary, long documents are underestimated, since they contain more terms, and the average frequency of the terms in the text below. To combat this effect is applied normalized frequency, which is calculated as 0.5 + 0.5(*TF/NTF*), where *NTF* is the average frequency of the term in the document. An alternative method of weighting terms in the texts of the document is the logarithm of the frequency of occurrence of the term. In this case, the weight of terms included in the text of the document is defined as 1+log (*TF*). To compensate for the effect of different resource lengths, a similar frequency normalization is used. In this case, the formula looks like $\frac{1+\log(TF)}{1+\log(MTF)}$, where *MTF* is the frequency of the word that occurs the most times in the document.

These methods for determining weight are well described in [12].

Today, a text document model based on the static measure TF-IDF is widely used.

Suppose we have a dictionary W - an ordered set of terms whose power is M. The power of a dictionary is the number of terms it contains. Then the document can be represented as a vector:

$$Di =< wi1, wi2, \dots wiM >, \quad (1)$$

where w_{ki} is the frequency of the k-th term in the i-th text document (i = (1, n)).

$$TF_{ki} = \frac{m_{ik}}{M_i}, \quad (2)$$

where m_{ki} is the number of occurrences of the *k*-th term in the *i*-th document M_i — the total number of terms in the i-th document.

$$IDF_k = ln\frac{N}{n_k}, \quad (3)$$

where N is the total number of documents in the collection, n_k is the number of documents in the collection in which the *k*-th term is found. The value IDF_k characterizes the importance of the *k*-th term in the collection of documents. The frequency of the term is calculated using the TF-IDF formula:

$$w_{ik} = TF_{ki} \times IDF_k. \quad (4)$$

6. Model of a text document with consideration of time

Information exhibits a temporal decay, losing its value and utility over time. This phenomenon arises from the continuous emergence of newer, more relevant, comprehensive, accurate, and reliable information.

When designing text analysis algorithms, it is crucial to consider the dynamic nature of information. By incorporating temporal factors, we can prioritize more recent and relevant documents.

The rate of information aging varies across different subject areas and topics. For instance, the evolution of mathematical concepts differs from the rapid advancements in information technology. Numerous factors influence this aging process, including the specific development trajectory of each topic. In 1960, R. Barton and R. Kebler conducted a study to determine the "half-life" of publications in various fields: physics (9.2 years), mathematics (21 years), and geology (23.6 years). Analogous to the concept in quantum mechanics, the "half-life" of a scientific article represents the period during which half of all publications on a given topic are released.

To obtain a quantitative estimate of the rate of aging of scientific publications R. Barton and R. Kebler used the Malthus model. This model may occur under the following assumptions:

- There is N - the number of scientific papers on some topics.
- It is assumed that the growth rate of the number of scientific works is directly proportional to N.

The latter assumption follows from the statistical studies of the international research firm IDC.

If we consider the model of the document, which is given by formulas (1-4), then one of the components of this model TF does not depend on time, provided that the number of words in the dictionary does not change with time. Indeed, TF is calculated according to formula (2), but neither the number of occurrences of the k-th term in the i-th document, nor the total number of terms in the i-th document change over time. The dynamic component of this model is the IDF value, which is calculated according to formula (3).

Let $IDF_k(t)$ be the importance of the *k*-th term in the collection of documents at time t on this topic. The relative change of this quantity over time Δt is defined as

$$R(t) = \frac{IDF_k(t + \Delta t) - IDF_k(t)}{IDF_k(t) * \Delta t}$$

Then

$$R(t) * IDF_k(t) = \frac{IDF_k(t + \Delta t) - IDF_k(t)}{\Delta t}$$

in passing to the limits we get:

$$R(t) * IDF_k(t) = \frac{dIDF_k(t)}{dt}$$

As a result, we obtained a differential equation with separable variables. Its solution depends on the function R(t), i.e. from the relative change in the importance of the k-th term in the collection of documents. Integrating the left and right sides of the equation:

$$\int_0^t \frac{dIDF_k}{IDF_k} = \int_0^t R(t)dt\ ,$$

$$\ln\left(IDF_k\right) = \int_0^t R(t)dt.$$

Or

$$IDF_k(t) = e^{\int_0^t R(t)\,dt}.$$

If R (t) is independent of time, i.e. is a constant, then we get the Malthus model:

$$IDF_k = IDF_{0k}e^{\alpha_c t - \beta_c t}$$

where IDF_{k0} - the importance of the k-th term at time t = 0, β_C - the half-life of the relevance of the document related to the subject C, is determined in an expert way, for each subject separately, α_C - growth rate of the relevance of the subject documents C, C- Subject text document.

The advantage of this model is that the Malthus equation has an exact solution in the form of a simple and convenient exponential function, but from the point of view of interpreting the results, it looks rather dubious. The main disadvantage of this approach is that the exponential function cannot describe events that have local extremes, but for a large number of text documents, the Malthus model is correct.

Consider a model (1), where the weight of the term w_{ik} is defined by formula (4). This formula is the product of the stationary component of the TF and the dynamic IDF. Then relying on the Malthus model, you can get:

$$w_{ik}(t) = TF_{ik} * IDF_{0k}e^{\alpha_c t - \beta_c t} \quad (5)$$

Thus, a dynamic model of a text document is obtained.

7. Results

Given a collection of text documents, the objective is to retrieve a subset of documents that are most relevant to a specific information need, expressed as a query. To facilitate this process, the document collection is mathematically represented as a matrix (1), where each row corresponds to a document vector. These document vectors are constructed using either model (4) or (5), which are likely statistical language models or semantic representations. Similarly, the query, which is typically a short text phrase, can also be represented as a vector using the same model.

Often, the same methods are used in the search engine search module as in the indexing module. For example, in the vector model, the search is built on the basis of the tuple $<W, L_D, S, Q>$, where W is the dictionary, that is, an ordered set of terms, the power of the dictionary is denoted as N; information flow is a matrix: $L_K = D_i$ of dimension KxN, where search images of K documents. Similarly, they present a vector to the query: $Q = \{q = <t_1,t_2,..t_N>\}$. Hence, the search procedure S has the form $Lxq=r$, where q is the query vector, r is the system response to the query.

In order to search for information on a given query, you need to sort through all the documents in the collection and calculate the distance between the vector representing the collection document and the vector representing the query. The smaller the distance between the document vector and the request vector, the more this document corresponds to the request:

$$r_{kq} = \sqrt{\sum_{i=1}^{n}(w_{ik} - w_{iq})^2},$$

where r_k - is the distance from the vector that represents the k-th document to the vector that represents the request, w_{ik} - the coordinates of the vector that represents the k-th document, w_{iq} - the coordinates of the vector that represents the request.

Further, the documents are sorted, depending on the distance to the request and the most relevant documents are selected.

Search quality characteristics are divided into two error levels. A first-level error is considered if the document is not mistakenly among the documents sought. Errors of the second level include errors when a document is mistakenly found in the searched documents. Let the number of documents in the test set be equal to N, of which N_p is the number of documents matching the request, and N_n is the number of documents that are not related to the request.

Then, $N = N_p + N_n$.

Let the number of false passes F_n, and false detections F_P, therefore the number of correct passes and correct detections: $T_P=N_p-F_n$; $T_n = N_n-F_P$. The degree of accuracy and completeness that are often used in information retrieval tasks is calculated based on the characteristics of T_P and F_P:

$$precision = P = \frac{T_p}{T_p+F_p} \times 100\%,$$

$$recall = R = \frac{T_p}{T_p + F_n}$$

Completeness measures the proportion of correct information across all documents. Accuracy measures the proportion of true detections of all identified resources. Completeness and accuracy are values dependent on each other. During the development of search engine architecture, one usually has to choose one of two characteristics as dominant. If the choice fell on accuracy, this leads to a decrease in completeness due to an increase in the number of false-positive answers. The increase in completeness causes a simultaneous drop in accuracy. Therefore, it is convenient to characterize the search engine using one value, the so-called F_1-measure or Van Riesbergen measure.

$$F_1 = 2\frac{P \times R}{P+R}.$$

Measure F_1 is one of the most common characteristics of such systems. There are two main approaches to calculating F_1 for text document search tasks: total F_1 (the results for all tests are summarized in one table, by which the measure F_1 is then calculated) and the average F_1 (for each test its own F_1 value is generated, then the arithmetic average for all tests).

The percentage of errors allows you to determine the correctness metric:

$$A = \frac{T_p+N_p}{N}.$$

To conduct the experiments, a corpus of 400 Ukrainian text documents was utilized. This corpus included 250 documents pertaining to the field of continuum mechanics. Given the rapid obsolescence of information in physics, with a half-life of approximately 4.6 years, the experiments focused on retrieving relevant documents within this specific domain.

The software implementation of these experiments was conducted using Python 3.8 within the Spider Integrated Development Environment. Text documents were processed using regular expressions and libraries such as re for regular expressions, pynlple for natural language processing, and pandas for data manipulation and analysis. Unnecessary meta-information was removed from the documents.

To perform the lemmatization, the pymorphy2 library. This library also includes Ukrainian morphological dictionaries, enabling lemmatization of Ukrainian text. Furthermore, pymorphy2 facilitates the removal of stop words, which are common words that contribute little to the semantic content of a text.

As a result of the work, the following results were obtained, shown in Fig. 2-4.

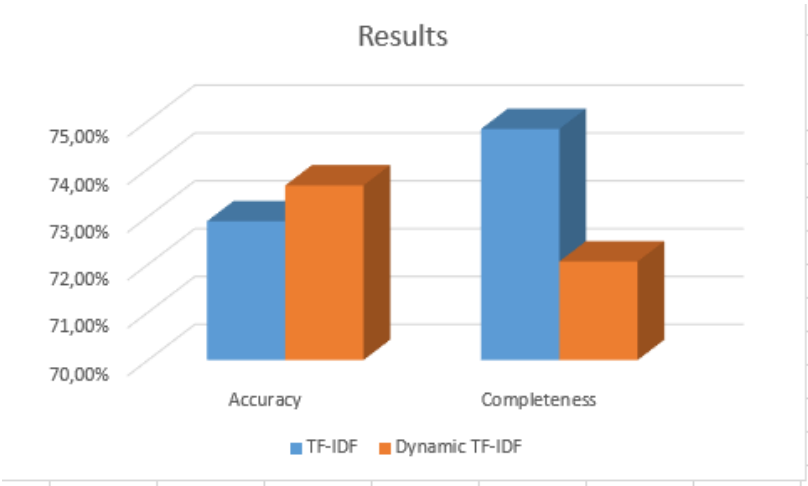

Figure 2: Accuracy and Completeness.

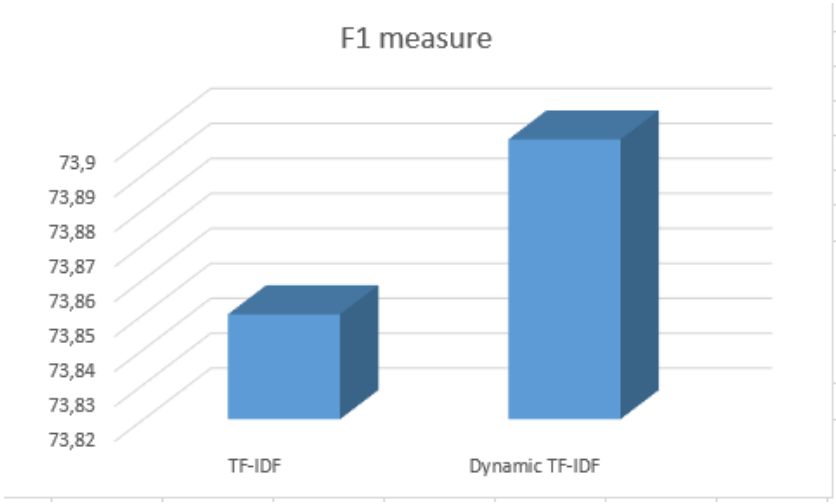

Figure 3: F1 measure.

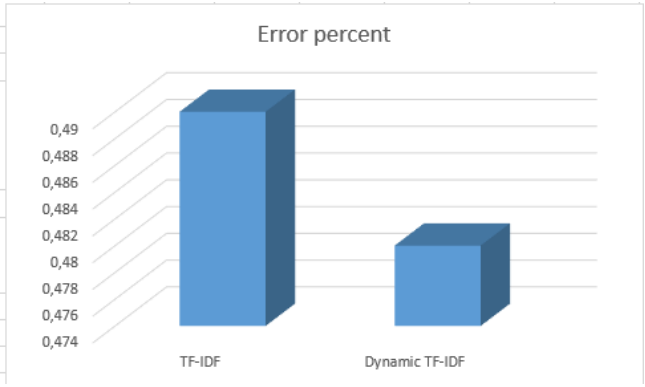

Figure 4: Error percent.

8. Conclusion

For searching at text documents, it is essential to consider their temporal dynamics, as the relevance of information changes over time. In particular, when searching for documents containing data for decision-making or process control, attention should be paid to the document's creation date.

The proposed text document classification model allows for the automation of this process. The inclusion of a temporal component in the model can refine the document's position in the classifier, reflecting its relevance. The Malthusian model provides an accurate solution for document retrieval in the form of an exponential function. However, this method has limitations when describing phenomena with local extrema. Nevertheless, for large volumes of text data, the Malthusian model is quite effective.

Experiments have shown that search accuracy increased by 0.8%, with a loss of search completeness of 3%.

Declaration on Generative AI

The authors have not employed any Generative AI tools.

References

[1] Z. Zhang, L. Chen, F. Yin, X. Zhang, L. Guo, Improving online clustering of Chinese technology web news with bag-of-near-synonyms. IEEE Access, 8, (2020).

[2] Z. Feng, X. Tian, A Scientific Document Retrieval and Reordering Method by Incorporating HFS and LSD, J. Applied Sciences, 13(20), (2023).
[3] R. Szczepanek, A deep learning model of spatial distance and named entity recognition (SD-NER) for flood mark text classification, Water, 15(6), (2023).
[4] Haytham, S. A. S., Lars, K. N. I. P. P. I. N. G., Carmen, P. E. T. C. U., Automatic Clustering of e-Commerce Product Description, J. Applied Computer Science & Mathematics, 6(2), (2011).
[5] Z. Zhang, L. Chen, F. Yin, X. Zhang, L. Guo, Improving online clustering of Chinese technology web news with bag-of-near-synonyms. IEEE Access, 8, (2020).
[6] C. F. Marton, Salton and Buckley's Landmark Research in Experimental Text Information Retrieval, (2011).
[7] J. Xu, A natural language processing based technique for sentiment analysis of college english corpus, J. Peer Computer Science, 9, (2023).
[8] I. Yurchuk, O. Gurnik, Tongue twisters detection in Ukrainian by using TDA, CEUR Workshop Proceedings 3396 (2023).
[9] T. Kovaluk, K. Dukhnovska, O. Kovtun, A. Nikolaienko, I. Yurchuk, Text classification using term co-occurrence matrix. XX International Scientific Conference "Dynamical System Modelling and Stability Investigation " (DSMSI-2023), December 19-21, 2023.
[10] Y. Kravchenko, O. Leshchenko, N. Dakhno, O. Pliushch, O. Trush and Y. Yermakov, "Development of Model of Artificial Ecosystem on the Basis of Genetic Algorithm," 2022 IEEE 4ht International Conference on Advanced Trends in Information Theory (ATIT), 2022, pp. 199–203. doi: 10.1109/ATIT58178.2022.10024214.
[11] H. Shevchenko, N. Dakhno, O. Leshchenko, O. Barabash, Y. Kravchenko, A. Dudnik, Using Mathematical Optimization Methods to Maximize Audience Reach with Budget Constraints. In 2022 IEEE 4th International Conference on Advanced Trends in Information Theory (ATIT), 2022, pp. 249-254. URL: https://www.scopus.com/inward/record.uri?eid=2-s2.0-85147915607&doi=10.1109%2fATIT58178.2022.10024187&partnerID=40&md5=74e4115b60e8750cdd5c8c877148d4d4 doi: 10.1109/ATIT58178.2022.10024187.
[12] M. Lapata, Automatic evaluation of information ordering: Kendall's tau. Computational Linguistics, 32(4), (2006).