applied sciences

MDPI

Article

Text Similarity Detection in Agglutinative Languages: A Case Study of Kazakh Using Hybrid N-Gram and Semantic Models

Svitlana Biloshchytska [1,2,*], Arailym Tleubayeva [3,*], Oleksandr Kuchanskyi [1,4,5,*], Andrii Biloshchytskyi [1,2], Yurii Andrashko [6], Sapar Toxanov [7], Aidos Mukhatayev [8] and Saltanat Sharipova [9]

[1] Department of Computational and Data Science, Astana IT University, Astana 010000, Kazakhstan; a.b@astanait.edu.kz
[2] Department of Information Technology, Kyiv National University of Construction and Architecture, 03037 Kyiv, Ukraine
[3] Department of Computer Engineering, Astana IT University, Astana 010000, Kazakhstan
[4] Department of Information Control Systems and Technologies, Uzhhorod National University, 88000 Uzhhorod, Ukraine
[5] Department of Biomedical Cybernetics, National Technical University of Ukraine "Igor Sikorsky Kyiv Polytechnic Institute", 03056 Kyiv, Ukraine
[6] Department of System Analysis and Optimization Theory, Uzhhorod National University, 88000 Uzhhorod, Ukraine; yurii.andrashko@uzhnu.edu.ua
[7] Department of Administration, Astana IT University, Astana 010000, Kazakhstan; sapar.toxanov@astanait.edu.kz
[8] Department of General Education Disciplines, Astana IT University, Astana 010000, Kazakhstan; aidos.mukhatayev@astanait.edu.kz
[9] Center of Competence and Excellence, Astana IT University, Astana 010000, Kazakhstan; saltanat.sharipova@astanait.edu.kz
* Correspondence: bsv@astanait.edu.kz (S.B.); a.tleubayeva@astanait.edu.kz (A.T.); kuchanskyi.o@gmail.com (O.K.)

Abstract: This study presents an advanced hybrid approach for detecting near-duplicate texts in the Kazakh language, addressing the specific challenges posed by its agglutinative morphology. The proposed method combines statistical and semantic techniques, including N-gram analysis, TF-IDF, LSH, LSA, and LDA, and is benchmarked against the bert-base-multilingual-cased model. Experiments were conducted on the purpose-built Arailym-aitu/KazakhTextDuplicates corpus, which contains over 25,000 manually modified text fragments using typical techniques, such as paraphrasing, word order changes, synonym substitution, and morphological transformations. The results show that the hybrid model achieves a precision of 1.00, a recall of 0.73, and an F1-score of 0.84, significantly outperforming traditional N-gram and TF-IDF approaches and demonstrating comparable accuracy to the BERT model while requiring substantially lower computational resources. The hybrid model proved highly effective in detecting various types of near-duplicate texts, including paraphrased and structurally modified content, making it suitable for practical applications in academic integrity verification, plagiarism detection, and intelligent text analysis. Moreover, this study highlights the potential of lightweight hybrid architectures as a practical alternative to large transformer-based models, particularly for languages with limited annotated corpora and linguistic resources. It lays the foundation for future research in cross-lingual duplicate detection and deep model adaptation for the Kazakh language.

Keywords: anti-plagiarism; Kazakh language; combined models; text data analysis; near duplicates; semantic analysis; academic integrity; intelligent analysis system

Academic Editor: Andrea Prati

Received: 2 May 2025
Revised: 9 June 2025
Accepted: 12 June 2025
Published: 15 June 2025

Citation: Biloshchytska, S.; Tleubayeva, A.; Kuchanskyi, O.; Biloshchytskyi, A.; Andrashko, Y.; Toxanov, S.; Mukhatayev, A.; Sharipova, S. Text Similarity Detection in Agglutinative Languages: A Case Study of Kazakh Using Hybrid N-Gram and Semantic Models. *Appl. Sci.* **2025**, *15*, 6707. https://doi.org/10.3390/app15126707

1. Introduction

In recent years, there has been a rapid increase in the volume of digital content, which demands increasingly accurate and efficient methods for originality verification. As of 2023, approximately 30% of academic articles and student papers in universities worldwide contain instances of borrowing, with 10% of these cases resulting from deliberate plagiarism [1]. In the context of global digitalization, the task of detecting not only direct copies but also near duplicates has become more relevant than ever. It is expected that by 2025, the global market for plagiarism detection software will exceed USD 2.5 billion, with combined data analysis models used for detecting near duplicates constituting a significant share. To protect intellectual property and maintain academic integrity, increasing attention is being paid to methods for detecting duplicate and similar texts. However, in certain languages, particularly agglutinative ones like Kazakh, the application of standard approaches may be less effective. Agglutination leads to the formation of long morphological chains, complicating the task of identifying matches at the token level [2,3]. This necessitates the investigation and comparison of various methods capable of accurately detecting near duplicates in the Kazakh language.

Near duplicates are text fragments that partially match the original but have differences such as paraphrasing, structural changes, or minor content modifications. For instance, paraphrasing and structural modification of text are the most common plagiarism techniques, accounting for up to 60% of all cases of borrowing [4]. Detecting such near duplicates in the Kazakh language requires more complex algorithms than standard methods for finding identical fragments. This is because traditional anti-plagiarism systems, based on simple text comparisons, often prove insufficiently effective at identifying subtle changes and partial borrowing.

Agglutination in the Kazakh language complicates text processing, as a single word can contain numerous suffixes and affixes that alter its lexical and grammatical function. Methods focused solely on matching individual lexemes may overlook semantic similarity. Hybrid approaches that account not only for statistical metrics (e.g., TF-IDF) but also for semantic components (e.g., latent semantic analysis (LSA)) enable a more accurate consideration of the diversity of morphological forms in the Kazakh language. Based on existing research, it can be stated that text data analysis, particularly for agglutinative languages, requires not only the application of standard methods but also their adaptation to the language's specific features. Previous studies demonstrate that traditional approaches, such as statistical methods or frequency-based analysis, can be useful; however, they often face limitations when processing texts with high morphological complexity. This highlights the need for more flexible solutions that consider both surface-level and deep-level textual similarity [5].

The main features of modern text data analysis systems that contribute to their popularity include accuracy and adaptability to various types of data. The use of advanced analysis methods, such as locality-sensitive hashing and N-gram analysis, enables the detection of text fragments with a high degree of similarity, even in the presence of significant text modifications. Studies show that combined models, which integrate statistical and semantic approaches can improve the accuracy of detecting near duplicates by 25% compared to traditional algorithms [6]. These algorithms include text preprocessing and indexing, which not only enhance accuracy but also significantly reduce analysis time—from several hours to minutes for a standard academic document. These anti-plagiarism systems play an important role not only in maintaining academic integrity but also in human resource management, as they help create transparent and objective conditions for evaluating the knowledge and skills of students and applicants. This helps educational institutions and employers to build trust with candidates and conduct fair employee certification. In the

context of education globalization, such systems contribute to uniform assessment standards, making educational programs more comparable at an international level. As a result, workforce mobility increases, and diplomas and certifications obtained in different countries become more interchangeable and recognized, facilitating the employment of graduates in the global labor market.

Kazakh (ISO 639-3, kaz) is a Kipchak (Northwestern) Turkic language with approximately ten million speakers [7,8]. While the majority of Kazakh speakers live in the Republic of Kazakhstan, significant Kazakh-speaking populations exist throughout Central Asia. Furthermore, it should be noted that text modeling and near-duplicate detection in the Kazakh language have several distinctive features related to its morphological, syntactic, and lexical characteristics. Specifically, Kazakh has agglutinative morphology, meaning that words can be formed by attaching numerous affixes to the root. This results in many word forms and challenges in tokenization, as standard text segmentation methods may be ineffective. In Kazakh, grammatical relationships are determined by affixes rather than word order, which complicates models based on word sequences (e.g., N-grams). Additionally, lemmatization in Kazakh is challenging because standard algorithms may fail to account for numerous suffixes. Stemming is also complicated, as it requires correctly distinguishing between the root word and all possible affixes.

Kazakh exhibits strong polysemy and synonyms, meaning that words can have multiple meanings depending on the context, and many concepts can be expressed using different words, making text classification and near-duplicate detection more difficult. The official and scientific styles in Kazakh often contain long and complex sentences, requiring specialized approaches to syntactic analysis and complicating text normalization. Moreover, an additional challenge is that Kazakh is a low-resource language. All these factors must be considered when developing and implementing hybrid methods for near-duplicate detection in text documents. The development of such methods for low-resource agglutinative languages has significant theoretical importance for the advancement of applied and corpus linguistics.

Additionally, the creation and implementation of these methods have substantial practical significance, particularly for the academic community in Kazakhstan. The experience gained from implementing hybrid near-duplicate detection methods is valuable not only for Kazakh linguistics but also for other Turkic languages (e.g., Kyrgyz, Turkish, Azerbaijani, Uzbek), Mongolic languages, Uralic languages (e.g., Finnish, Estonian, Hungarian), as well as Korean and Japanese, which feature agglutinative morphology and complex affix/postfix systems.

Thus, the development of combined models for text data analysis opens up new opportunities for enhancing academic integrity and preventing plagiarism. It is forecasted that the implementation of such methods will improve verification reliability by 30% and reduce cases of unintentional borrowing in academic articles and dissertations by more than 40% [9]. Thus, the development of hybrid methods for identifying near duplicates, considering the characteristics of agglutinative and low-resource languages, including Kazakh, is highly relevant.

2. Literature Review

The analysis of the theoretical aspects of detecting near duplicates and combating plagiarism shows that this task requires comprehensive and multi-level approaches. In recent years, researchers have proposed numerous methods and models aimed at improving the accuracy and reliability of text analysis in the context of ever-increasing volumes of digital content in the Kazakh language. Modern studies focus on creating flexible models that can

effectively recognize both complete and near duplicates, which is especially important in the academic environment, where maintaining standards of academic integrity is crucial.

Hybrid approaches combining syntactic and semantic analysis have improved detection accuracy. Locality-sensitive hashing and N-gram analysis are effective for identifying similarities in modified text fragments [10–13]. In particular, the methods for identifying near duplicates in scientific papers, which include the content of the same type, such as text data, mathematical formulas, and numerical data, are described in the article [14]. For text data, the method of locally sensitive hashing, which involves the finding of Hamming distance between the elements of indices of scientific papers was formalized. For numerical data, subsequences for each scientific work are formed, and the proximity between the papers is determined as the Euclidian distance between the vectors consisting of the numbers of these subsequences. To compare mathematical formulas, the method of comparing sample formulas is used and the names of variables are compared. Two directions are separated: finding key points in the image and applying locally sensitive hashing for individual pixels of the image to identify near duplicates in graphic information. Lizunov et al. [12] describe the application of detecting near duplicates in tables using the locality-sensitive hashing method and the nearest neighbor method. However, it should be noted that these studies largely depend on the collected database and do not consider the peculiarities of agglutinative languages, particularly the Kazakh language.

LSA and probabilistic topic models enhance thematic similarity analysis and support the assessment of academic work, particularly in dissertations and theses [14–19]. Detecting near duplicates in scientific papers is a complex task, as research materials contain various types of data, including text, mathematical formulas, images, diagrams, charts, and numerical data. For the practical application of near-duplicate detection methods, it is necessary to identify the scientific subject spaces to which the research materials belong. This can be achieved using LSA, as described in [14]. Based on these methods, information systems can be developed to identify near duplicates in scientific texts and documents. Biloshchytskyi et al. [19] present a conceptual model for building such a system, which considers the detection of near duplicates across various data types. However, this model does not consider the specific challenges of detecting near duplicates in agglutinative and low-resource languages.

Studies of Kazakh texts show that combining statistical methods with structural analysis improves phrase extraction accuracy. For instance, bigrams and trigrams help identify candidate phrases, while rule-based methods achieve higher accuracy for basic noun groups [20]. Similarly, classification tasks for Kazakh social media texts, such as identifying extremist content, benefit from TF-IDF, N-gram methods, and LSTM models, demonstrating high classification efficiency [21]. Bakiyev proposes an extension of the TF-IDF method for the Kazakh language, which considers synonyms when determining the similarity of textual documents [22]. This method improves the accuracy of document similarity measurement, which is crucial for plagiarism detection and text clustering tasks. Considering the linguistic features of the Kazakh language is an important aspect of ensuring the accuracy of near-duplicate identification. Specifically, Akanova et al. [23] develop a keyword search algorithm for a corpus of Kazakh texts, incorporating the language's morphological characteristics and utilizing neural networks to enhance search accuracy. Rakhimova et al. propose a hybrid approach to semantic text analysis in the Kazakh language, combining statistical and linguistic methods [5]. This allows consideration of the agglutinative nature of Kazakh and improves the accuracy of text analysis. Kosyak and Tyers [24] examine various segmentation methods to improve language modeling and text prediction for low-resource and agglutinative languages, including Kazakh. However, from the perspective of applied linguistics, developing hybrid methods for near-duplicate detection that account for the

specific features of the Kazakh language remains an important and promising direction. It is important to recognize that, in addition to text, documents can contain a variety of data types, including images, mathematical formulas, numerical data organized into tables, and more. The detection of multimedia borrowings, such as images and diagrams, requires specialized methods. Techniques such as keypoint-based algorithms (SIFT, SURF) and perceptual hashing enable the identification of modified visuals [25–27]. Deep learning models, including CNNs and RNNs, further enhance plagiarism detection by analyzing complex textual and graphical data [28].

Hybrid methods that integrate TF-IDF with semantic understanding, such as the TSWT structure and HowNet database, outperform traditional algorithms in text similarity tasks, improving precision and recall [29]. Advanced anti-plagiarism systems also assess citation quality and originality levels, guiding researchers and students to produce higher-quality, original work [30]. It is particularly interesting to explore the performance of hybrid methods for near-duplicate identification in agglutinative and low-resource languages. Based on a review of well-known methods, a comparative analysis was conducted, and the main advantages and disadvantages of the near-duplicate detection methods were identified; they are presented in Table 1.

Table 1. Main advantages and disadvantages of the near-duplicate detection methods.

Method	Description	Advantages	Disadvantages
N-gram	Comparison of overlapping N-grams (typically using cosine similarity)	Very simple and fast computation	Sensitive to word order and synonyms
TF-IDF	Document–term vectorization + cosine similarity	Captures statistical similarity	High number of false positives
Hybrid	N-gram Jaccard + LSA + LDA, weighted combination	Balance of statistical and semantic features, flexible tuning	Requires multiple processing stages, more complex
BERT	bert-base-multilingual-cased with cosine similarity	Acceptable semantic accuracy	Difficult to set up; resource-intensive (GPU/CPU)

Kazakh is an agglutinative Turkic language where grammatical meanings are expressed via sequences of suffixes. This leads to a large number of word forms. By some estimates, there are over 750 affixes [31], which complicates morphological analysis and hinders direct comparison of texts. Semantically equivalent words may appear in different surface forms, which can reduce the accuracy of lexical matching. Kazakh has a relatively free word order (default SOV) thanks to rich case morphology, which marks syntactic roles [32]. This creates challenges for methods based on word order, as similar texts may differ in word sequence.

Additionally, Kazakh lacks grammatical gender and articles; definiteness is marked contextually or via case endings. Verbs are inflected for tense, mood, person, and number, often eliminating the need for personal pronouns. In contrast, English relies on a fixed word order and auxiliary words (articles, prepositions, etc.) [33]. Lexically, Kazakh exhibits high synonymy and polysemy, featuring numerous loanwords alongside native terms. As a result, semantic analysis requires vector-based models that can recognize meaning equivalence across different word choices. Overall, effective comparison of Kazakh texts requires a hybrid approach: morphological normalization, syntactic parsing, and semantic models resilient to paraphrasing and lexical variability.

Let us consider an example that illustrates the difference. In Kazakh, the sentence "Студент университетте кітапты оқыды" is equivalent to the sentence "The student read the book at the university" in English. In Kazakh, case endings allow flexible word order "Кітапты студент университетте оқыды" without altering the meaning. In English, word order is crucial; reordering the words as "The book the student at the university read" results in an ungrammatical sentence. Table 2 describes the features of the Kazakh language in comparison with English.

Table 2. The features of the Kazakh language in comparison with English.

Parameter	Kazakh Language	English Language
Language type	Agglutinative	Analytic
Morphology	Rich suffix-based inflection	Limited inflectional morphology
Word order	Flexible (SOV)	Fixed (SVO)
Articles and gender	No articles or grammatical gender	Articles present, gender in pronouns
Verb system	Complex affixation	Analytic tense/aspect forms
Lexical features	High synonymy and polysemy	More standardized vocabulary

The analysis of theoretical aspects and approaches to detecting plagiarism and near duplicates in Kazakh texts demonstrates that this issue holds significant importance for education and human resource management. Thus, the research highlights various approaches to text similarity analysis, including statistical, semantic, and their combinations. Integrating structural and semantic information helps to improve accuracy and adapt methods to the specific features of languages and tasks. Considering semantics and optimizing features, selection proves beneficial when working with texts in morphologically complex languages such as Kazakh. These methods can be applied to tasks such as detecting borrowings, text classification, and semantic analysis.

The literature review indicates the need to address the scientific gap related to the lack of effective and linguistically adapted methods for detecting near duplicates in low-resource agglutinative languages, such as Kazakh, by developing and validating a hybrid model that combines statistical and semantic analysis. Thus, the goal of this study is to develop a hybrid model for near-duplicate detection and to conduct a comparative analysis of four approaches to identifying textual similarities: N-gram, TF-IDF, BERT, and a hybrid model combining statistical and semantic analysis. The objectives include describing and formalizing near-duplicate identification according to the hybrid model, creating a Kazakh language corpus for conducting experiments, and evaluating the effectiveness of the hybrid model for near-duplicate detection, as well as the N-gram, BERT, and TF-IDF methods, for the Kazakh language case study.

3. Materials and Methods

The methodology of this study focuses on the development and testing of combined models for text data analysis aimed at detecting near duplicates in academic and scientific documents written in Kazakh language. The primary emphasis is on creating a hybrid system that integrates several approaches, including N-gram analysis, locality-sensitive hashing (LSH), LSA, and probabilistic topic models. The first stage of the research involved collecting data from various sources, including scientific articles, theses, term papers, and dissertations, all of which are available in open-access databases in the Kazakh language. The dataset included both original documents and works with partially modified sections.

The combined model developed for text analysis incorporates several key components. N-gram analysis was used to identify matches at the phrase level and detect paraphrased text fragments, with bigrams and trigrams tested for optimal accuracy. LSH was employed for the rapid detection of similar text fragments, allowing the identification of partially modified sections and significantly speeding up data processing. LSA was applied for the semantic comparison of texts, identifying conceptual similarity between fragments, which is useful for detecting hidden plagiarism. The probabilistic topic model (PTM) facilitated the analysis of the thematic structure of the text and the evaluation of the completeness of topic representation in publications, which is particularly important for assessing the thematic alignment of scientific publications.

The model was trained on the collected data, including both original and borrowed texts. Although the term cross-validation is used here in a broad sense, we did not apply a classical k-fold cross-validation procedure, since LSA and LDA do not require iterative optimization via gradient descent. Instead, the corpus was randomly split into an 80% training set and a 20% test set. From the training set, an additional 20% hold-out subset was used for tuning threshold hyperparameters. The final evaluation of model performance (precision, recall, F1-score) was conducted on the independent test set. Performance evaluation was conducted using precision, recall, and F1-score metrics, enabling an objective assessment of the model's effectiveness in detecting near duplicates. Various parameters were tested for each method, including the N-gram size and the number of topics, to optimize model settings and improve accuracy. Model validation included comparisons with N-gram, TF-IDF, analysis of false positives, and assessment of processing time, which ultimately refined and enhanced the quality of the combined model's performance.

The results of testing the models on prepared datasets enabled a comparative analysis of their performance and revealed the specific features of each method when applied to Kazakh language texts. The proposed approach ensures the adaptation of existing methods to the linguistic characteristics of the language, making them more suitable for text similarity analysis tasks.

It is known that documents can contain data of various types: text, images, tables, mathematical formulas, diagrams, and charts. For each of these data types, specialized analysis methods need to be applied. Image analysis in documents can be outperformed using both standard software applications and additional custom modules. To improve recognition quality, additional processing methods may be required, such as restoring blurred images, analyzing shape and texture, extracting image fragments, and matching two images based on key points, among others. Most image recognition methods are based on keypoint detection, image hashing, stochastic geometry, Markov chains, perceptual hashing methods, and others. These methods are highly effective in identifying near-image duplicates. All image recognition methods can be categorized: methods based on key point extraction and methods based on pixel-level image analysis.

Before applying recognition methods, it is necessary to filter graphical data, which simplifies and increases the sensitivity of analysis methods for detecting similarities in these data. The characteristic feature of filtering methods is that they allow the extraction of specific characteristics of local image regions. Filtering involves applying transformations to the entire image, such as binarization with a given threshold, high-frequency filtering (Gabor filter), low-frequency filtering (Gaussian filter), Fourier filtering, and more. A specific type of image filtering involves identifying elementary mathematical functions within the image (such as lines, parabolas, circles, etc.). Filters of this type include Hough, Radon, and other filters. In the case of complex images with many fragments, it is often sufficient to analyze not the entire image but only its contours.

To formally cover all possible cases and data types for identifying near duplicates, the table format will be considered. Since tables may contain textual information, numerical data, mathematical formulas, and images, if only some of these data types need to be analyzed, the other types are simply disregarded.

Let B be some input table, and $\overline{B} = \{B_1, B_2, \ldots, B_p\}$ be a set of tables extracted from the text and stored in a common table database, where p is the number of tables in the database. The task is to identify a set of tables $\widetilde{B} = \left\{\widetilde{B}_1, \widetilde{B}_2, \ldots, \widetilde{B}_l\right\}$ where $\widetilde{B} \subset \overline{B}$, $l < p$, for which the condition holds: $F\left(B, \widetilde{B}_i\right) < \lambda$, $i = \overline{1,l}$, where λ is threshold value, and F is the distance between the tables. Distance as a metric must satisfy the properties of non-negativity, the identity of indiscernible factors, symmetry, and triangle inequality. The presence in the database of at least one table from the set $\overline{B}$, for which the condition $F\left(B, \widetilde{B}_i\right) < \lambda$ is satisfied, indicates that table B is borrowed [12].

A cell K_{ij} of some table B is an element of the table that is formed at the intersection of the i-th row and the j-th column of this table. The cell content K_{ij} of table B is the value (numeric, text, date type, etc.) corresponding to this cell. Let us denote the content of cell K_{ij} as $C(K_{ij})$, where $i = \overline{1, r_1}$, $j = \overline{1, r_2}$ where r_1 is the number of rows of table B, and r_2 is the number of columns of table B. As previously defined, the content of cells may include the following types: numeric, text, date, image, formula, combined data. If the table contains images and formulas, they are allocated for separate analysis. The content of the "date" type after conversion to a single format may be considered as a regular text string.

Let $I = \{1, 2, \ldots, r_1\}$ be a set of table row numbers, and $J = \{1, 2, \ldots, r_2\}$ be a set of column numbers of a table B. We iterate through all the cells of the table and determine their data type. Thus, simplifying the possible variants of cell content types, we can consider that the content of the table is presented as $B = \langle \overline{N}, \overline{S} \rangle$. If the data in a cell belongs to the numeric type, then the corresponding content is presented as an element of the set $\overline{N}$; if to the text type, then it is added to the set $\overline{S}$ according to the following rule:

$$\overline{N} = \left\{ k \middle| k \in C(K_{ij}), k \in R, i \in I, j \in J \right\}, \tag{1}$$

$$\overline{S} = \left\{ k \middle| k \in C(K_{ij}), k \in T, i \in I, j \in J \right\}, \tag{2}$$

where set T is a set of symbols, $L = \mathrm{card}(T)$,

$$T = \{t_1, t_2, \ldots, t_L\}, \tag{3}$$

t_i is a separate symbol, $t_i \in A$, A is the set of symbols.

We represent sets $\overline{N}$ and $\overline{S}$ in the form of a numerical sequence and a sequence of strings, respectively, of length v and w. Thus, $N = \{n_1, n_2, \ldots, n_v\}$ is the sequence of numeric values of the cell content, $v = \mathrm{card}(N)$, $S = \{s_1, s_2, \ldots, s_w\}$ is the sequence of cell content rows, and $w = \mathrm{card}(S)$. Let us define separately the representation of these sequences in a form convenient for applying models of similarity identification and searching for near duplicates.

Define the sequence $S = \{s_1, s_2, \ldots, s_w\}$. Each of its elements contains text, which can consist of one or more words. By sequentially viewing all the elements of the sequence from s_1 to s_w, let us select words from them. The word of an arbitrary element—a unigram of the sequence S—is given as a sequence:

$$S_n^{\beta} = \{t_1, t_2, \ldots, t_{\beta}\}, \tag{4}$$

where $n \in \mathbb{N}$ is an ordinal number of a word, β is word length, and $t_j \in A$, $t_j \notin C$, $j = \overline{1,\beta}$, $C = \{\text{“\_”}, \text{“,”}, \text{“.”}, \text{“ ”}, \text{“ − ”}, \text{“ : ”}, \text{“;”}, \text{“\#”}, \ldots\}$ are all non-letter characters of the sequence elements, $S_n^{\beta} \in s_j$, $j \in \{1, 2, \ldots, w\}$.

A new unigram is defined consisting of all words from the original unigram of strings S, after removing the so-called stop words. The list of stop words is specified as a set $M = \{\text{“and”}, \text{“or”}, \text{“but”}, \text{“in”}, \text{“on”}, \text{“at”}, \text{“with”}, \ldots\}$ or the same words. Then the new sequence of words has the following form:

$$W = \left\{S_1^{\beta_1}, S_2^{\beta_2}, \ldots, S_m^{\beta_m}\right\}, \tag{5}$$

where β_j, $j = \overline{1,m}$ are the word lengths, and m is their number. The elements of such a sequence are words in canonical form.

Using the sliding window method, we construct a set of sequences:

$$\begin{gathered} E_1 = \left\{S_1^{\beta_1}, S_2^{\beta_2}, \ldots, S_h^{\beta_h}\right\}, \\ E_2 = \left\{S_2^{\beta_2}, S_3^{\beta_3}, \ldots, S_{h+1}^{\beta_{h+1}}\right\}, \\ \ldots \\ E_{m-h+1} = \left\{S_{m-h+1}^{\beta_{m-h+1}}, S_{m-h+2}^{\beta_{m-h+2}}, \ldots, S_{m-1}^{\beta_{m-1}}, S_m^{\beta_m}\right\}, \end{gathered} \tag{6}$$

where h is the window size or number of elements of constructed sequences $E_1, E_2, \ldots, E_{m-h+1}$.

Next, using the locality-sensitive hashing method, we represent the set of sequences $F(W) = (E_1, E_2, \ldots, E_{m-h+1})$ in the form of bit strings:

$$\Delta(W) = (I(E_1), I(E_2), \ldots, I(E_{m-h+1})), \tag{7}$$

where $I(E_k)$ is the index element that specifies a bit string that uniquely corresponds to a sequence E_k, $k = \overline{1, m-h+1}$. Thus

$$I(E_k) = \{\delta_{k1}, \delta_{k2}, \ldots, \delta_{kc}\}, \tag{8}$$

where $\delta_{kx} \in \{0, 1\}$, $k = \overline{1, m-h+1}$, $x = \overline{1,c}$, and c is the number of bits that make up a bit sequence.

Let us consider the numerical sequence $N = \{n_1, n_2, \ldots, n_v\}$ of input table B and construct a set of subsequences for it using the sliding window method:

$$\begin{gathered} K_1 = \left\{n_1, n_2, \ldots, n_g\right\}, \\ K_2 = \left\{n_2, n_3, \ldots, n_{g+1}\right\}, \\ \ldots \\ K_{v-g+1} = \left\{n_{v-g+1}, n_{v-g+2}, \ldots, n_{v-1}, n_v\right\}, \end{gathered} \tag{9}$$

where v is the number of elements in a sequence N, and g is the window size or number of elements in subsequences $K_1, K_2, \ldots, K_{v-g+1}$. Since the elements of the constructed subsequences are real numbers, $n_i \in R$, $i = \overline{1,v}$, then these subsequences can be considered as g-dimensional vectors. If we assume that the space is given R^g with a Euclidean structure, then we can define a metric on this space ρ between any two vectors of space $a \in R^g$ and $b \in R^g$: $\rho(a,b)$. In this case, this metric will satisfy the axioms of identity, that is, $\rho(a,b) = 0 \Leftrightarrow a = b$, axiom of symmetry: $\rho(a,b) = \rho(b,a)$ and the triangle axiom for some vector $c \in R^g$: $\rho(a,c) \leq \rho(a,b) + \rho(b,c)$.

Such a distance or similarity measure between the vectors representing the numerical values of the content of the tables is a component that determines the degree of similarity of these tables.

In accordance with the statement of the problem, let B be the input table, and $B_1, B_2, \ldots, B_p$ be the tables that are selected and stored in a common table base, where p is the number of tables in the base. The task is to identify those tables from the base for which the condition $F\left(B, \widetilde{B}_i\right) < \lambda$, $i = \overline{1,l}$ is satisfied. Let a sequence of text data be constructed, $S = \{s_1, s_2, \ldots, s_w\}$, and a sequence of numerical values $N = \{n_1, n_2, \ldots, n_v\}$ for the table B. Since the base of the tables is already known, it is obvious that each such table is indexed. Thus, for each of the tables $B_1, B_2, \ldots, B_p$ there are known sequences of text data $S^y = \{s_1^y, s_2^y, \ldots, s_w^y\}$ and sequences of numerical data $N^y = \{n_1^y, n_2^y, \ldots, n_v^y\}$, $y = \overline{1,p}$.

It is also obvious that index elements are given for word sequences:

$$I\left(E_{k_y}^y\right) = \left\{\delta_{k_y1}^y, \delta_{k_y2}^y, \ldots, \delta_{k_yc}^y\right\}, \tag{10}$$

where $\delta_{k_yx}^y \in \{0,1\}$, $k_y = \overline{1, m_y - h + 1}$, $x = \overline{1,c}$, c is the number of bits in a bit sequence, and m_y is the number of words in the sequences:

$$W^y = \left\{S_1^{y,\beta_1}, S_2^{y,\beta_2}, \ldots, S_{m_y}^{y,\beta_{m_y}}\right\} \tag{11}$$

containing words S_y^{y,β_y} in canonized form, where β_y, $j = \overline{1, m_y}$ is word lengths.

We construct a string sequence $S = \{s_1, s_2, \ldots, s_w\}$ input table B unigram of words in canonized form:

$$W = \left\{S_1^{\beta_1}, S_2^{\beta_2}, \ldots, S_m^{\beta_m}\right\}, \tag{12}$$

where β_j, $j = \overline{1,m}$ is word lengths, and m is their number. Next, using the sliding window method, we determine the sequences $E_1, E_2, \ldots, E_{m-h+1}$ and, using the locality-sensitive hashing method, we construct the index elements:

$$I(E_k) = \{\delta_{k1}, \delta_{k2}, \ldots, \delta_{kc}\}, \tag{13}$$

where $\delta_{kx} \in \{0,1\}$, $k = \overline{1, m - h + 1}$, $x = \overline{1,c}$, and c is the number of bits representing the sequence.

We calculate the Hamming distances between the elements of each index of the input table sequences and the elements of the indexes of the table sequences in the database using the formula:

$$H\left(I(E_k), I\left(E_{k_y}^y\right)\right) = \frac{1}{c}\sum_{j=1}^{c}\left|\delta_{kj} - \delta_{k_yj}\right|, \tag{14}$$

where $k = \overline{1, m - h + 1}$, $k_y = \overline{1, m_y - h + 1}$, $y = \overline{1,p}$.

Given that

$$H\left(I(E_k), I\left(E_{k_y}^y\right)\right) < \lambda_H, \tag{15}$$

where $\lambda_H \in [0,1]$ is a predetermined parameter value, with probability one; it can be stated that the index element with number k is similar to an index element with a number k_y tables with number y. Consequently, the table with the number y can be similar to an input table with a threshold λ_H, that is, it contains a near duplicate.

We construct a finite numerical sequence $N = \{n_1, n_2, \ldots, n_v\}$ input table B set of subsequences $K_1, K_2, \ldots, K_{v-g+1}$. We also consider that for each of the tables $B_1, B_2, \ldots, B_p$,

based on their numerical data sequences $N^y = \{n_1^y, n_2^y, \ldots, n_v^y\}$, $y = \overline{1,p}$, subsequences are constructed as $K_1^y, K_2^y, \ldots, K_{v-g+1}^y$ by the sliding window method:

$$\begin{gathered} K_1^y = \{n_1^y, n_2^y, \ldots, n_g^y\}, \\ K_2^y = \left\{n_2^y, n_3^y, \ldots, n_{g+1}^y\right\}, \\ \ldots \\ K_{v-g+1}^y = \left\{n_{v-g+1}^y, n_{v-g+2}^y, \ldots, n_{v-1}^y, n_v^y\right\}. \end{gathered} \tag{16}$$

If we represent the constructed subsequences $K_1, K_2, \ldots, K_{v-g+1}$ and $K_1^y, K_2^y, \ldots, K_{v-g+1}^y$ in the form of tuples, then their similarity measures are determined based on the Euclidean distance, the urban metric, or the Minkowski distance. Then for each table a distance matrix will be:

$$\rho_1\left(K_u, K_r^y\right) = \sqrt{\sum_{j=r}^{g+r-1}\left(n_{j+u-r} - n_j^y\right)^2}, \tag{17}$$

$$\rho_2\left(K_u, K_r^y\right) = \sum_{j=r}^{g+r-1}\left|n_{j+u-r} - n_j^y\right|, \tag{18}$$

$$\rho_3\left(K_u, K_r^y\right) = \left(\sum_{j=r}^{g+r-1}\left|n_{j+u-r} - n_j^y\right|^t\right)^{\frac{1}{t}}, \tag{19}$$

$y = \overline{1,p}$, $u = \overline{1, v-g+1}$, $r = \overline{1, v-g+1}$, t is the Minkowski distance parameter.

Then for each $y = \overline{1,p}$ we find the minimum values for each row of matrices $\rho_\tau\left(K_u, K_r^y\right)$, and we get the distances:

$$\xi_\tau\left(K_u, K_{min}^y\right) = \min_{r=\overline{1,v-g+1}}\left\{\rho_\tau\left(K_u, K_r^y\right)\right\}, \tag{20}$$

$u = \overline{1, v-g+1}$ for fixed $\tau = \overline{1,3}$.

If the condition is met $\xi_\tau\left(K_u, K_{min}^y\right) < \lambda_p$ for a predetermined parameter value $\lambda_p \in [0,1]$, then it can be asserted with probability one that the vector with number u is similar to a vector of a table with a number y, that is, the table contains a near duplicate. The larger the value λ_p, the more stringent requirements are imposed on the search for near duplicates. The distance for $y = \overline{1,p}$, $u = \overline{1, v-g+1}$ is normalized:

$$\xi_\tau^N\left(K_u, K_{min}^y\right) = \frac{\xi_\tau\left(K_u, K_{min}^y\right) - \min\limits_{u=\overline{1,v-g+1}}\left\{\xi_\tau\left(K_u, K_{min}^y\right)\right\}}{\max\limits_{u=\overline{1,v-g+1}}\left\{\xi_\tau\left(K_u, K_{min}^y\right)\right\} - \min\limits_{u=\overline{1,v-g+1}}\left\{\xi_\tau\left(K_u, K_{min}^y\right)\right\}}. \tag{21}$$

Figure 1 illustrates how the hybrid model works for detecting similar text fragments and tables. First, the system receives input data. Then, it cleans the text by removing unnecessary characters and words, splits it into parts, and extracts values from tables. After that, the data is processed by different modules: one checks matches based on N-grams, another evaluates semantic similarity, and a third analyzes numerical values in the tables. All the results are then combined in a single module, where they are compared against a threshold to determine whether the fragments are duplicates or not.

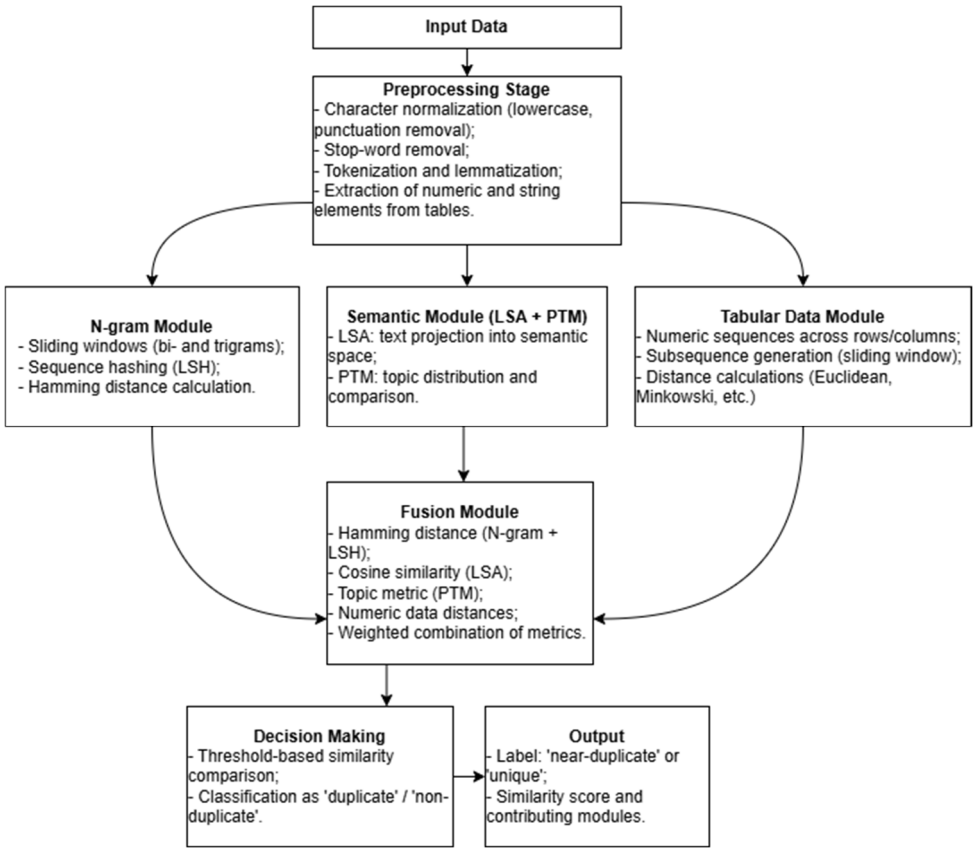

Figure 1. Diagram of the hybrid method.

The described method allows the finding of near duplicates in tables considering the text and numeric representation of data. Similarly, using this method, it is possible to index numeric and text data if they are not in a table, but in the main content of a document. The proposed hybrid method for detecting near duplicates in tables allows the identification of similarities in text and numeric data and then the generalization of the results. For text data, sequences are created in canonical form, which are converted into bit sequences based on hashing. Similarity is calculated by the Hamming distance with a given threshold. Similarity in numeric data is determined based on the nearest neighbor method, which allows the identification of near duplicates relative to a set of tables selected from scientific publications and dissertations. It should be noted that the method is designed to search for near duplicates in tables containing only text and numeric data. In the case of images and formulas, they are analyzed separately using specific methods.

4. Case Study: The Dataset for Duplicate Detection in Kazakh

The Arailym-aitu/KazakhTextDuplicates corpus is designed to identify close duplicates in Kazakh language texts and is based on the Arailym-tleubayeva/small_kazakh_corpus dataset [34], which contains Kazakh Wikipedia articles. A total of 25,922 text fragments are extracted from this set, providing a variety of topics and style variants. To simulate real-life scenarios of partial borrowing, typical for student papers, scientific publications, and internet content, a system of controlled modifications of source texts was used to obtain pairs of texts with different types of similarity.

The set of transformations aimed at preserving the main semantic content while changing the form of the text was used to generate duplicates. In particular, the following methods were applied:

1. Word and phrase rearrangement. The order of words in sentences was changed (e.g., rearranging homogeneous members or fragments of complex sentences) without compromising grammatical correctness and meaning, thus mimicking a change in the narrative order while preserving the content.
2. Synonym substitution. Keywords were replaced with synonymous equivalents using external lexical resources (electronic thesauri and synonym dictionaries) and, where necessary, manual analysis, considering part of speech and context, thus preserving the original meaning.
3. Morphological changes. Changes in the grammatical form of words (number, tense, case) were made without changing their lexical meaning, e.g., replacing the active voice with the passive voice or converting direct speech into indirect speech, using morphological analysis and synthesis tools to ensure correct agreement.

In addition, the technique of partial deletion or addition of textual fragments was used to model partial duplicates, where individual sentences were either deleted from the original paragraph or added with new sentences that did not alter the general subject matter, thereby imitating situations of fragmentary borrowing.

Each pair of texts in the corpus was classified into four types of duplicates, depending on the nature of the changes made and the degree of preservation of the original content:

1. Exact duplicate. The texts are virtually identical except for minor edits (e.g., case or punctuation changes).
2. Partial duplicate. Only a significant portion of the text is the same, but the remainder may be modified or supplemented.
3. Paraphrase duplicate. Texts express the same information through paraphrasing, which involves replacing vocabulary with synonyms and modifying syntax, resulting in varied wording while retaining the key meaning.
4. Contextual duplicate: texts do not explicitly match at the phrase level, but describe a similar context, plot, or idea, which is characteristic of cases where borrowing occurs at the level of an idea or narrative structure.

At the preprocessing stage, the texts were normalized, which included normalizing all characters to lowercase and removing extra spaces and special characters, as well as filtering stop words («және», «бірақ», «мен», «сен», «ол», «бұл», «соң», etc.), the list of which was generated based on common sets for information retrieval tasks and manually adjusted to the specifics of the corpus. When generating paraphrases, stop words were not replaced by synonyms, thus avoiding artificial increase in similarity between texts.

The KazakhTextDuplicates corpus covers a wide range of textual borrowing scenarios, including direct copying to deep paraphrasing and hidden contextual paraphrasing. Being based on the Arailym-tleubayeva/small_kazakh_corpus set, it represents a unique resource for training and testing algorithms for automatic plagiarism detection, adapting existing methods to consider the linguistic peculiarities of the Kazakh language and improving the efficiency of anti-plagiarism systems under resource-limited conditions.

Main characteristics of the corpus (Table 3):

- Total size: 217 MB (before conversion);
- Size in Parquet format: 96.2 MB;
- Number of lines (text fragments): 25,922;
- Data formats: CSV, Parquet.

The texts were deliberately modified using the following methods: word rearrangement, using synonyms, adding or removing morphological suffixes, and replacing collocations with similar constructions to form the test sample.

Table 3. Dataset characteristics.

Type_Duplicate	Count
exact	6597
partial	6468
paraphrase	6439
contextual	6418

The purpose of the corpus is to model real-world scenarios of partial borrowing, which is an urgent task in the fields of text duplicate detection and anti-plagiarism. This approach allows a comprehensive evaluation of the effectiveness of algorithms for detecting exact, contextual, and partial duplicates in Kazakh language texts and contributes to the development of methods for processing and analyzing textual data in low-resource languages.

5. Results

We conducted an experiment aimed at comparative evaluation of four approaches to detecting close duplicates in Kazakh texts: N-gram analysis, the TF-IDF method, the base BERT model, and the proposed hybrid model combining semantic and syntactic analysis. The following key metrics were used to comprehensively evaluate the performance of each method: precision (Precision), completeness (Recall), F1-measure, and processing time.

The dataset Arailym-aitu/KazakhTextDuplicates, formed based on Arailym-tleubayeva/small_kazakh_corpusn [34], includes 25,922 text fragments covering a wide range of topics and style variants, which ensures its representativeness for modeling real scenarios of partial borrowing. The analysis shows that the system of controlled modifications, including word rearrangements, replacement of key terms with synonyms and morphological transformations, allows the efficient generation of pairs of texts classified into exact, partial, paraphrase and contextual types, which reflects the diversity of ways of processing the source material. The normalization of texts by lower-casing, removal of redundant characters, and stop word filtering helps to reduce the impact of minor variations, which improves the accuracy of the markup. The obtained dataset demonstrates high quality of markup and homogeneity of data, which makes it a valuable resource for the development and testing of plagiarism detection algorithms and the adaptation of existing methods to the peculiarities of the agglutinative and low-resource Kazakh language.

Standard statistical metrics, such as precision, recall, F1-score, and processing time are used to quantify the performance of algorithms for detecting textual duplicates in the KazakhTextDuplicates dataset. Precision is defined as the proportion of correctly identified duplicates in relation to the total number of detected duplicates, which allows us to assess the propensity of the model to produce false positives; recall measures the proportion of correctly found duplicates in relation to the total number of true borrowings, demonstrating the ability of the algorithm to detect all relevant cases; F1-score, being a harmonic average of precision and recall, serves to assess the balance of the model; processing time characterizes the computational efficiency and applicability of the methods when working with large datasets The complex analysis of these indicators allows us to objectively compare the performance of different approaches (such as N-gram, TF-IDF, and the hybrid model) and optimize their parameters in order to improve the accuracy and reliability of automatic plagiarism detection systems in low-resource Kazakh.

At the preprocessing stage, all the texts were lower-cased and punctuation marks and stop words were removed to reduce noise and ensure a uniform presentation of the data.

The results of the N-gram analysis depended significantly on the chosen value of n. Unigrams achieved perfect accuracy (1.00) and completeness (1.00), but the lack of context consideration limited the detection of paraphrased or partially modified texts. When mov-

ing to bigrams and trigrams, there is a decrease in completeness to 0.51 while maintaining accuracy at 1.00, resulting in an F1-measure of 0.67. At the same time processing time increases from 14.81 s (bigrams) to 16.79 s (trigrams) (see Table 4 and Figure 2).

Table 4. Comparison of N-gram for different values of n.

N-gram (n)	Precision	Recall	F1-Score	Processing Time (s)
1 (Unigram)	1.00	1.00	1.00	19.55
2 (Bigram)	1.00	0.51	0.67	14.81
3 (Trigram)	1.00	0.51	0.67	16.79

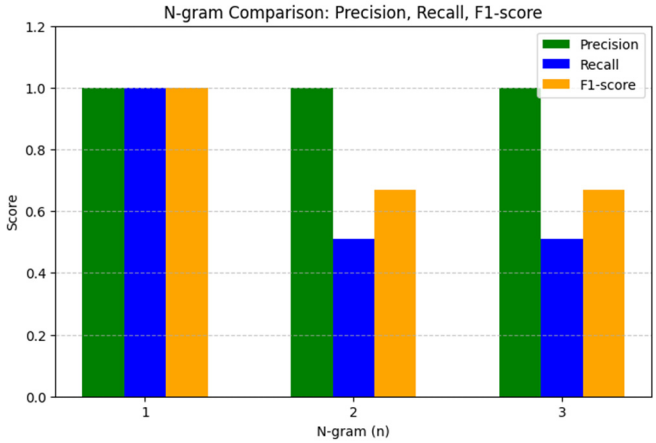

Figure 2. N-gram comparison of metrics.

Thus, N-gram analysis effectively detects exact matches, but fails with paraphrased or partially modified texts. The TF-IDF model showed an F1-measure of 0.67, which is comparable to the N-gram approach, but with a different balancing of metrics: accuracy was 0.51 and completeness was 1.00. The high completeness indicates the model's ability to detect most modified texts, but the reduced accuracy indicates a higher number of false positives. The processing time of TF-IDF was the most efficient at 12.81 s.

The dataset was divided into training, validation, and test subsets. A small validation subset (20% of the training data) was separated from the training corpus and used to tune threshold values. The main part of the training set was used solely for training LSA, LDA, and constructing MinHash representations, while the test set was used for final evaluation. During the experiments, special attention was paid to ensuring that fragments from the same source document did not appear in both training and test sets simultaneously. To guarantee no overlap, a group-based split was used (GroupKFold by source document ID).

In the hybrid method using LSA, LDA, and MinHash, there is no classical "training" phase involving gradient descent. However, there is a parameter-fitting stage (e.g., fitting SVD for LSA, training LDA models, and generating hashes), which was carried out only on the training data. The hybrid approach involved multiple threshold values (λ) that affected the final decision: thresholds for Jaccard similarity (MinHash), for cosine similarity in the LSA space, for cosine similarity in the LDA space, and a final aggregated threshold λ_{fused}. Each of these thresholds was treated as a hyperparameter, as their values directly influenced the model's precision, recall, and ultimately the F1-score. A grid search over combinations of these thresholds was conducted on the validation subset. As a result, the optimal final threshold was determined to be $\lambda_{\text{fused}} = 0.7$, which provided the best trade-off: precision ≈ 1.00 (no false positives), recall ≈ 0.73 (still a high detection rate), F1-score ≈ 0.84.

Setting a lower λ_{fused} value (e.g., 0.5) resulted in a drop in precision to ~0.50, which is unacceptable for industrial systems, where each false positive requires manual review. On the other hand, setting $\lambda_{fused} \geq 0.8$ led to a recall below 0.65, indicating an unacceptably low level of detection.

Below are Table 5 and Figure 3 summarizing the comparison of metrics for the N-gram, TF-IDF, and hybrid models. Figure 3 presents a comparison of the models in terms of execution time on the validation set.

Table 5. Comparison of metrics for N-gram, TF-IDF, base BERT, and hybrid models.

Model	Precision	Recall	F1-Score	Processing Time (s)
N-gram (n = 3)	1.00	0.51	0.67	16.79
TF-IDF	0.51	1.00	0.67	12.81
Hybrid model	1.00	0.73	0.84	0.87
bert-base-multilingual-cased	1.00	0.49	0.66	104.39

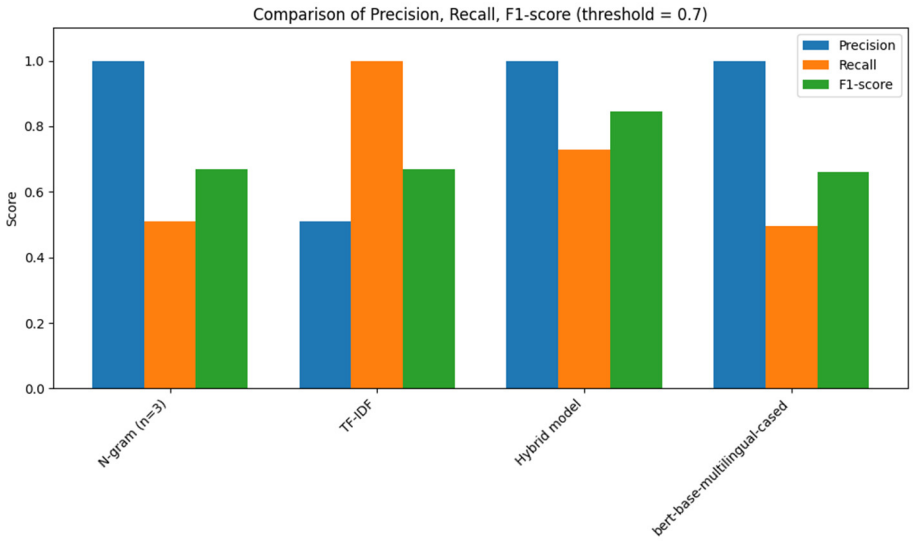

Figure 3. Comparison of precision, recall, F1-score.

As shown in Table 3, although all models yield a comparable F1-score of around 0.67, they differ in terms of precision, recall, and processing time. The N-gram model offers high precision but lower recall, whereas TF-IDF demonstrates high recall with lower precision. The hybrid model maintains high recall that is similar to that of TF-IDF but at the expense of significantly increased computational cost.

We conducted a series of experiments with the threshold value and confirmed that increasing λ_{fused} to 0.7 completely eliminates false positives on the test set while maintaining an acceptable detection level (Table 3). Thus, by accepting a minimal reduction in recall, we achieved perfect precision, making the system highly reliable for automated text verification tasks where false positives are critical.

To justify the relevance and competitiveness of the hybrid approach, we conducted a direct comparison with the baseline BERT-like model bert-base-multilingual-cased [35]. The proposed hybrid method demonstrates comparable accuracy while requiring significantly fewer computational resources and avoiding the need for large pre-trained transformer models. This confirms that the combination of lightweight models remains a practical and efficient solution for near-duplicate detection tasks in low-resource agglutinative languages.

6. Discussion and Conclusions

This paper presents a hybrid method for detecting near duplicates in tables represented in documents in the Kazakh language. By using locally sensitive hashing and the nearest neighbor method, the proposed approach enables effective identification of similarities between text and numerical data, which is crucial for ensuring academic integrity and detecting potential plagiarism. Through the integration of hash methods and Hamming distance for textual content, as well as the use of metric distances for numerical data, this method can even identify partially matching data.

The developed algorithm adapts to textual, numerical, and mixed data, ensuring accurate analysis of complex tables through standardization and normalization, reducing false positives. The threshold values for matching enhance precision across various data types. A comparative analysis of N-gram, TF-IDF, BERT, and the hybrid model reveals that the hybrid model outperforms the others in the terms of precision, recall, and F1-score. Its balance of statistical and semantic methods enables efficient large-scale text similarity analysis, particularly for Kazakh texts. For the task of near-duplicate detection, achieving precision = 1 is a strong result, as it indicates the complete absence of false positives. This is particularly important because if unique texts are incorrectly identified as similar to others without clearly establishing partial duplication, the plagiarism detection process becomes ineffective. Therefore, in our case, the model's performance will primarily be evaluated based on recall and F1 score, as they better reflect the system's ability to identify actual duplicates without compromising reliability.

The hybrid method for near-duplicate detection aligns with the specifics of agglutinative languages due to its consideration of morphological variability through normalization and semantic generalization, free word order through order-insensitive metrics, and synonymy and polysemy through thematic and conceptual similarity.

Future work should focus on developing embeddings for Kazakh and adapting deep language models, such as BERT, to improve accuracy. Testing on cross-lingual datasets could address challenges posed by diverse morphological structures. The method's flexibility makes it highly applicable in plagiarism detection and intelligent data analysis for Kazakh language texts, especially in academic and research contexts.

Since it was very important to achieve zero false positives (FP = 0), the resulting precision remained consistently perfect (1.00) across all hold-out/test splits. Notably, even under a classical cross-validation scheme, the false positive rate would remain zero across all folds, leading to an error bar of zero for the precision metric. This strict performance was attained by applying a conservative threshold of $\lambda_{\text{fused}} = 0.7$. Under this setting, the model rejects any candidate pair in which at least one of the three similarity signals (MinHash, LSA, or LDA) does not exceed the specified threshold. This design reflects a conscious trade-off, prioritizing absolute certainty in positive predictions (high precision) at the expense of a moderate reduction in recall (approximately 0.73).

The combination of statistical and semantic approaches provides a more comprehensive detection of textual duplicates in Kazakh texts compared to traditional methods such as N-gram analysis and TF-IDF. Although all the methods considered show a comparable F1-measure value (around 0.67), there are significant differences in the distribution of accuracy and completeness, as well as in the processing time.

The analysis reveals that the N-gram approach is characterized by high accuracy, but its limited completeness indicates an inability to detect cases of paraphrasing or partial modifications. The TF-IDF model, on the other hand, exhibits high completeness, which allows efficient capture of modified content, but false positives occur at the cost of reduced accuracy. The hybrid model, which combines both approaches, provides high completeness similar to TF-IDF while showing moderate accuracy. However, a significant disadvantage

of the hybrid method is the significantly longer processing time due to the additional computational complexity of integrating semantic and syntactic analyses.

The practical applicability of each method depends on the specifics of the task. For applications where processing speed is critical, the TF-IDF method may be preferred, whereas for tasks requiring comprehensive and detailed identification of different types of borrowings, the hybrid model is a more suitable choice, despite the increased computational cost.

In the future, it is advisable to conduct further research on the integration of specialized embeddings for the Kazakh language, as well as the adaptation of deep language models such as BERT, which will improve detection quality by taking deeper account of semantic and contextual features of the text. Additionally, extending testing to cross-lingual datasets will help address the problems associated with the diversity of morphological structures in related languages.

Thus, the proposed hybrid model is a promising solution for textual duplicate detection in low-resource agglutinative language contexts, thereby contributing to the improvement of accuracy and reliability in academic integrity monitoring and textual data mining systems.

Author Contributions: Conceptualization, S.B., A.T. and O.K.; methodology, O.K. and A.T.; software, A.T. and Y.A.; validation, A.B., A.T. and Y.A.; formal analysis, S.T.; investigation, S.B. and A.B.; writing—original draft preparation, A.T., O.K. and S.T.; writing—review and editing, A.M. and S.S.; visualization, A.T.; supervision, A.B.; project administration, S.B. All authors have read and agreed to the published version of the manuscript.

Funding: This paper was written in the framework of the state order to implement the research project, IRN No. AP23490123 «Development of a system to detect plagiarism using combined methods, models for finding near-duplicate, focusing on the Kazakh language».

Institutional Review Board Statement: Not applicable.

Informed Consent Statement: Not applicable.

Data Availability Statement: Tleubayeva, A. (2025). KazakhTextDuplicates: a dataset for duplicate detection in Kazakh. Hugging Face. https://huggingface.co/datasets/Arailym-aitu/KazakhTextDuplicates (accessed on 1 May 2025).

Conflicts of Interest: The authors declare no conflicts of interest. The funders had no role in the design of the study; in the collection, analyses, or interpretation of data; in the writing of the manuscript; or in the decision to publish the results.

References

1. Park, J.; Park, K. Analysis of the Causes and Types of Plagiarism in Academic Research. *J. Ethics High. Educ.* **2020**, *15*, 45–59.
2. Xu, J.; Pan, J.; Yan, Y. Agglutinative language speech recognition using automatic allophone deriving. *Chin. J. Electron.* **2016**, *25*, 328–333. [CrossRef]
3. Mammadov, S. Neural Spelling Correction for Azerbaijani Language. In Proceedings of the IEEE 13th International Conference on Application of Information and Communication Technologies, Azerbaijan, Baku, 23–25 October 2019; pp. 1–9.
4. Caroll, J. Student Plagiarism in Higher Education: A Systematic Review of Causes, Types, and Strategies for Detection. *Int. J. Acad. Integr.* **2019**, *10*, 223–237.
5. Rakhimova, D.; Turarbek, A.; Kopbosyn, L. Hybrid approach for the semantic analysis of texts in the Kazakh language. In Proceedings of the Asian Conference on Intelligent Information and Database Systems, Phuket, Thailand, 7–10 April 2021; Springer: Singapore, 2021; pp. 134–145.
6. Gupta, A.; Singhal, S. Hybrid Text Analysis Models for Improved Plagiarism Detection in Academic Texts. *Comput. Sci. Educ. J.* **2021**, *20*, 78–90.
7. Mukhamedova, R. *Kazakh: A Comprehensive Grammar*, 1st ed.; Routledge: Oxford, UK, 2015.
8. McCollum, A.; Chen, S. Kazakh. *J. Int. Phon. Assoc.* **2021**, *51*, 276–298. [CrossRef]
9. Smith, R. Advances in Plagiarism Detection Techniques: Trends and Impacts in Academic Publishing. *J. Acad. Ethics* **2023**, *18*, 101–115.

10. Varol, C.; Hari, S. Detecting near-duplicate text documents with a hybrid approach. *Comput. J. Inf. Sci.* **2015**, *41*, 405–414. [CrossRef]
11. Pagani, F.; Dell'Amico, M.; Balzarotti, D. Beyond precision and recall: Understanding uses (and misuses) of similarity hashes in binary analysis. In Proceedings of the Eighth ACM Conference on Data and Application Security and Privacy, Tempe, AZ, USA, 19–21 March 2018; pp. 354–365.
12. Lizunov, P.; Biloshchytskyi, A.; Kuchansky, A.; Biloshchytska, S.; Chala, L. Detection of near duplicates in tables based on the locality-sensitive hashing method and the nearest neighbor method. *Eur. J. Enterp. Technol.* **2016**, *6*, 4–10.
13. Lizunov, P.; Biloshchytskyi, A.; Kuchansky, A.; Andrashko, Y.; Biloshchytska, S.; Serbin, O. Development of the combined method of identification of near duplicates in electronic scientific works. *East.-Eur. J. Enterp. Technol.* **2021**, *4*, 57–63.
14. Lizunov, P.; Biloshchytskyi, A.; Kuchansky, A.; Andrashko, Y.; Biloshchytska, S. The use of probabilistic latent semantic analysis to identify scientific subject spaces and to evaluate the completeness of covering the results of dissertation studies. *East.-Eur. J. Enterp. Technol.* **2020**, *4*, 21–28. [CrossRef]
15. Watanabe, K. Latent semantic scaling: A semisupervised text analysis technique for new domains and languages. *Commun. Methods Meas.* **2021**, *15*, 81–102. [CrossRef]
16. Hassani, A.; Iranmanesh, A.; Mansouri, N. Text mining using nonnegative matrix factorization and latent semantic analysis. *Neural Comput. Appl.* **2021**, *33*, 13745–13766. [CrossRef]
17. Valdez, D.; Pickett, A.C.; Goodson, P. Topic modeling: Latent semantic analysis for the social sciences. *Soc. Sci. Q.* **2018**, *99*, 1665–1679. [CrossRef]
18. Qurashi, A.W.; Holmes, V.; Johnson, A.P. Document processing: Methods for semantic text similarity analysis. In Proceedings of the 2020 International Conference on INnovations in Intelligent SysTems and Applications (INISTA), Novi Sad, Serbia, 24–26 August 2020; pp. 1–6.
19. Biloshchytskyi, A.; Kuchansky, A.; Biloshchytska, S.; Dubnytska, A. Conceptual model of automatic system of near duplicates detection in electronic documents. In Proceedings of the 2017 14th International Conference The Experience of Designing and Application of CAD Systems in Microelectronics (CADSM), Lviv, Ukraine, 21–25 February 2017; pp. 381–384.
20. Altenbek, G.; Sun, R. Kazakh noun phrase extraction based on n-gram and rules. In Proceedings of the International Conference on Asian Language Processing, Harbin, China, 28–30 December 2010; pp. 305–308.
21. Mussiraliyeva, S.; Aslanbekkyzy, B.; Bolatkyzy, B. Investigating long short-term memory approach for extremist messages detection in Kazakh language. *Expert Syst.* **2024**, *42*, e13595. [CrossRef]
22. Bakiyev, B. Method for determining the similarity of text documents for the Kazakh language, taking Into account synonyms: Extension to TF-IDF. In Proceedings of the 2022 International Conference on Smart Information Systems and Technologies (SIST), Nur-Sultan, Kazakhstan, 28–30 April 2022; pp. 1–6.
23. Akanova, A.; Ospanova, N.; Kukharenko, Y.; Abildinova, G. Development of the algorithm of keyword search in the Kazakh language text corpus. *East.-Eur. J. Enterp. Technol.* **2019**, *5*, 26–32. [CrossRef]
24. Kosyak, S.; Tyers, F. Predictive Text for Agglutinative and Polysynthetic Languages. In Proceedings of the first workshop on NLP applications to field linguistics, Gyeongju, Republic of Korea, 12–17 October 2022; pp. 77–85.
25. Huang, Z.; Liu, S. Perceptual image hashing with texture and invariant vector distance for copy detection. *IEEE Trans. Multimed.* **2020**, *23*, 1516–1529. [CrossRef]
26. Thyagharajan, K.K.; Kalaiarasi, G. A review on near-duplicate detection of images using computer vision techniques. *Arch. Comput. Methods Eng.* **2021**, *28*, 897–916. [CrossRef]
27. Landge, A.; Mane, P. Near duplicate image matching techniques. In Proceedings of the 2016 International Conference on Information Communication and Embedded Systems (ICICES), Chennai, India, 25–26 February 2016; pp. 1–5.
28. Bayguzhina, A.; Sadybekova, G. The Use of Deep Neural Networks for Analyzing Images and Diagrams in Student Works. *J. Inf. Technol. Sci. Kazakhstan* **2022**, *15*, 56–70.
29. Fei, L. Research on Text Similarity Measurement Hybrid Algorithm with Term Semantic Information and TF-IDF Method. *Adv. Multimed.* **2022**, 7923262.
30. Meuschke, N.; Stange, V.; Schubotz, M.; Kramer, M.; Gipp, B. Improving Academic Plagiarism Detection for STEM Documents by Analyzing Mathematical Content and Citations. In Proceedings of the 2019 ACM/IEEE Joint Conference on Digital Libraries (JCDL), Champaign, IL, USA, 2–6 June 2019; pp. 120–129.
31. Fedotov, A.; Tussupov, J.; Sambetbayeva, M.; Idrisova, I.; Yerimbetova, A. Development and implementation of morphological model of Kazakh language. *Eurasian J. Math. Comput. Appl.* **2015**, *3*, 69–79.
32. Altenbek, G.; Wang, X.; Haisha, G. Identification of basic phrases for Kazakh language using maximum entropy model. In Proceedings of the COLING, Dublin, Ireland, 23–29 August 2014; pp. 1007–1014.
33. Dryer, M.S.; Haspelmath, M. *The World Atlas of Language Structures Online*; (v2020.4) [Data set]. Zenodo; Max Planck Institute for Evolutionary Anthropology: Leipzig, Germany, 2013. [CrossRef]

34. Tleubayeva, A. Small Kazakh Corpus. Hugging Face. 2025. Available online: https://huggingface.co/datasets/Arailym-tleubayeva/small_kazakh_corpus (accessed on 13 March 2025).
35. google-bert/bert-base-multilingual-cased. Hugging Face. Available online: https://huggingface.co/google-bert/bert-base-multilingual-cased (accessed on 11 March 2025).