DOI: https://doi.org/10.15276/aait.08.2025.4
UDC 004.91

Hybrid detection of fuzzy duplicate texts: cosine similarity and transformers

Tetiana M. Zabolotnia[1)]
ORCID: https://orcid.org/0000-0001-8570-7571; tetiana.zabolotnia@gmail.com. Scopus Author ID: 6507406568
Nazarii V. Kozynets[1)]
ORCID: https://orcid.org/0009-0009-1316-8340; kozinets.nazarii@gmail.com
[1)] National Technical University of Ukraine "Igor Sikorsky Kyiv Polytechnic Institute", 37, Beresteiskyi Ave. Kyiv, 03056, Ukraine

ABSTRACT

This paper addresses the challenge of detecting texts that share the same meaning but differ in wording and structure. Such "fuzzy duplicates" are increasingly prevalent in user-generated content, media articles, and academic materials. Traditional TF-IDF-based methods with cosine similarity process data swiftly but often overlook deeper semantic nuances, especially in languages with free word order and complex morphology (for example, Slavic languages such as Ukrainian or Bulgarian, and agglutinative languages like Hungarian). Fully neural solutions (e.g., transformers) typically offer higher accuracy yet can be slow and computationally demanding. To tackle these issues, we propose a hybrid approach that integrates a simplified neural component with classical cosine similarity. The workflow normalizes text variants (correcting spelling and inflectional forms), converts them into semantic vectors using a lightweight transformer model, and then applies a dynamic threshold mechanism tuned to text genre (e.g., news vs. social media). Experiments on Ukrainian-language datasets suggest that this method balances accuracy and speed more effectively than a fully neural pipeline. The approach is novel in combining domain-specific preprocessing and lightweight neural embeddings for fuzzy duplicate detection in text, achieving approximately ten to twelve percent higher detection accuracy than known solutions while maintaining faster runtime than a full BERT model. Preliminary tests in editorial and plagiarism-checking scenarios indicate that the system more reliably identifies paraphrased content than purely statistical methods, thereby reducing the burden of manual verification. Overall, the hybrid design offers a practical compromise between detection performance and computational requirements, which is especially beneficial for resource-constrained applications in morphologically rich languages like Ukrainian or other Slavic languages. Future efforts will focus on extending morphological coverage to further improve reliability.

Keywords: Hybrid methods; fuzzy duplicates; cosine similarity; transformer models; ukrainian language texts; content moderation systems

***For citation*: Zabolotia T. M., Kozynets N. V. "Hybrid detection of fuzzy duplicates: cosine similarity and transformers". *Applied Aspects of Information Technology*. 2025; Vol.8 No.1: 48–61. DOI: https://doi.org/10.15276/aait.08.2025.4**

INTRODUCTION

In the modern information environment, the volume of textual data is expanding at a rapid pace, creating significant challenges for efficient analysis and management. Fuzzy duplicates – texts that differ in wording yet express essentially the same meaning – complicate search engine indexing, distort analytics outcomes, and add to the strain on information storage systems. Traditional approaches, such as TF-IDF combined with cosine similarity, handle large datasets quickly but often ignore deeper semantic aspects, particularly in languages with complex morphology and free word order (for example, Slavic languages such as Ukrainian or Bulgarian, and agglutinative languages like Hungarian). Conversely, contemporary neural networks (e.g., BERT) offer higher accuracy but demand substantial computational resources. This situation motivates the exploration of hybrid solutions that unite the speed of classical methods with the contextual precision of neural embeddings.

The purpose of this paper is to develop a hybrid method for detecting fuzzy duplicates, combining classical similarity metrics (cosine similarity) with a lightweight transformer model (DistilBERT). Special attention is paid to Ukrainian-language data, given that morphological variations and syntactic flexibility can undermine the reliability of purely statistical or purely neural techniques in local information systems. Notably, these linguistic characteristics – free word order and rich morphology – are common across many languages (for instance, Slavic languages like Bulgarian or Polish, as well as Hungarian), so the proposed method is applicable beyond Ukrainian texts.

In this work, we present a hybrid method for fuzzy duplicate detection that merges specialized DistilBERT, fine-tuned for Ukrainian, with a cosine-based assessment of text pairs. By combining domain-focused preprocessing and lightweight embeddings, we enable robust semantic matching while limiting computational overhead. Additionally, we propose a dynamic thresholding

step, powered by a Random Forest [1] classifier, to handle varied text types such as formal news articles and informal social media posts.

The paper is organized as follows: the "Literature review and problem statement" section analyzes traditional and modern approaches to duplicate detection; the "Experimental Design and Methodological Approach" section describes the architecture of the hybrid method; the "Performance evaluation of the hybrid detection model" section compares the method's effectiveness with baseline solutions; the "Achieving practical balance in ukrainian and bulgarian text analysis" section evaluates the advantages and limitations of the approach; and the "Conclusions" section summarizes the findings and suggests future directions, such as exploring domain-specific tuning.

LITERATURE REVIEW AND PROBLEM STATEMENT

In recent years, information technology methods have been applied across various domains to solve specialized data management problems [2, 3], [4, 5], [6]. From electronic libraries to social media monitoring, these techniques help address issues like data redundancy and the distortion of analytic results. However, in the domain of fuzzy duplicate detection, existing approaches generally fall into two major categories: classical text similarity methods and modern neural network-based techniques.

1. Classical text similarity methods

The foundation for identifying near-duplicate textual content has historically relied on statistical algorithms, such as the TF-IDF (Term Frequency–Inverse Document Frequency) model coupled with cosine similarity [7]. In essence, TF-IDF treats each text as a high-dimensional vector, where each dimension corresponds to a unique word weighted by its frequency in the document and inversely by its frequency in the entire corpus.

Formally, the TF-IDF score for a term t in document d can be defined as:

$$TF\_IDF(t,d) = tf(t,d) \times \log\left(\frac{N}{df(t)}\right), \quad (1)$$

where N is the total number of documents, and $df(t)$ is the number of documents containing term t.

The cosine similarity then measures how closely two documents align in this vector space:

$$similarity = \cos(\theta) = \frac{A \cdot B}{\|A\|\|B\|}, \quad (2)$$

While this approach processes large sets of texts at relatively high speed, it has key shortcomings:

- semantic blindness: TF-IDF treats words in isolation, ignoring nuances like context, synonyms, and polysemy. Two sentences with the same meaning may share few lexical items, causing them to score low on similarity [8]. This problem becomes more pronounced in morphologically rich languages, including Ukrainian and Bulgarian, where words like "книга" vs. "книжку" (different grammatical forms of "book") and "книга" vs. "книгата" (Bulgarian, base vs. definite form) are viewed as unrelated [9];
- noise sensitivity: social media texts often contain abbreviations (e.g., "імхо") or slang forms (e.g., "норм"), which TF-IDF fails to normalize [10]. As a result, short posts with minimal shared vocabulary can be incorrectly flagged as dissimilar;
- scalability vs. accuracy trade-off: TF-IDF is computationally efficient for large corpora (e.g., 10,000 texts) [11], but its accuracy plateaus on subtle paraphrasing tasks. Studies on Ukrainian news corpora show F1-scores hovering at 0.65-0.74, significantly below human inter-rater agreement [12]. Alternate classical metrics – like the Jaccard Index or Levenshtein Distance – face similar barriers. The Jaccard Index, for instance, measures set overlap of tokens but cannot grasp that "кот на килимі" ("a cat on a rug") and "кіт лежить на підлозі" ("a cat lies on the floor") convey related ideas, despite no token overlap [13].

For instance, some studies enhance TF-IDF by expanding synonyms, incorporating phrase-level matching, or normalizing abbreviations. These improvements yield better recall for paraphrases, but they add complexity and still miss many semantic nuances. Such limitations highlight the need for context-sensitive algorithms.

2. Modern neural network-based approaches

With the introduction of transformer models, especially BERT (Bidirectional Encoder Representations from Transformers) [14, 15], semantic analysis of text underwent a profound transformation. Rather than treating words as discrete tokens with frequency-based weights, BERT employs a multi-layer bidirectional architecture with self-attention, capturing long-range contextual relationships. For instance, in Ukrainian, sentences like "Він працював у банку" ("He worked at a bank") and "Він грав на банку з водою" ("He tapped on a jar of water") are lexically similar but semantically distinct. BERT can usually

discern the difference via contextual embeddings [16].

Key developments include:

- sentence-BERT (SBERT): optimized for sentence-level embeddings, SBERT maps texts to a shared vector space, reducing the computational cost of pairwise comparisons from $O(N^2)$ to $O(N)$ [17];
- multilingual BERT (mBERT): provides cross-lingual embeddings, offering partial support for low-resource languages like Ukrainian. However, research shows a 15-20 % accuracy gap between Ukrainian and English performance, partly due to limited pretraining data [18];
- domain-focused Fine-Tuning: adapting BERT-like models to specific fields (legal, medical, etc.) often almost gives an increase in F1 scores [11].

Despite these benefits, neural approaches have drawbacks:

- high computational requirements: calculating pairwise similarity for thousands of text pairs can be time-consuming if the model is invoked repeatedly [19, 20];
- data hunger: large-scale annotated corpora remain scarce for Ukrainian. The UA-Corpus [21] is smaller than many English resources;
- morphological Complexity: ukrainian's inflectional grammar (e.g., "читатимуть" = "will read") strains subword tokenizers [22].

Recent efforts, such as compressing BERT into DistilBERT, improved speed but led to moderate accuracy drops, illustrating a trade-off between efficiency and precision [23].

3. Hybrid solutions in Natural Language Processing

Hybrid approaches in natural language processing strive to merge the computational efficiency of classical methods with the contextual advantages of neural networks, aiming to overcome the unique hurdles posed by morphologically rich languages like Ukrainian. A notable direction is the fusion of neural embeddings with statistical metrics. For example, some works compare the effectiveness of TF-IDF and Sentence-BERT in text processing tasks. One such study highlights that TF-IDF efficiently represents term importance within a corpus but lacks semantic understanding, whereas BERT-based models leverage contextual embeddings to capture meaning beyond surface-level word occurrences. The study finds that while TF-IDF performs well for keyword-based retrieval, BERT significantly improves semantic similarity tasks by recognizing paraphrases and contextual relationships more effectively. However, it also emphasizes that BERT requires more resources [24]. Recent efforts have explored ways to improve word embeddings for Ukrainian by fine-tuning FastText hyperparameters. A comparative study found that adjusting these parameters significantly impacts model performance across different text domains. For instance, optimized FastText embeddings demonstrated F1-scores of approximately 0.81 on formal content, such as news articles, while struggling with informal social media texts, where accuracy dropped to around 0.68 due to increased morphological variability and slang usage [25]. These findings highlight the need for further adaptation of word embeddings to better handle the linguistic diversity of Ukrainian. Another noteworthy pipeline employs a two-stage arrangement: BERT for initial candidate retrieval followed by a classical measure (e.g., Jaccard Index) for the final decision, achieving a 35 % reduction in runtime on a legal document set. Despite these creative designs, progress remains constrained by the lack of robust morphological tools for Ukrainian, such as advanced lemmatizers or tokenizers finely tuned to inflectional grammar [9].

To mitigate computational bottlenecks, model compression has garnered attention. Techniques like quantization, converting 32-bit model parameters into 8-bit integers, can shrink memory requirements by up to 75 % while causing only a modest 2–3% decline in accuracy [26]. Applying this to Ukrainian offers a potential path toward resource-friendly deployment, albeit with the caveat that morphological intricacies are not fully resolved. Alongside quantization, knowledge distillation – such as in MiniLM – transfers semantic capacity from a large model to a smaller one, retaining roughly 95 % of performance [27]. In a reported Ukrainian case study, a distilled variant of DistilBERT processed around 1,000 texts per second on a consumer-grade GPU – much higher than the rate for a full BERT model. However, these speed benefits often come with trade-offs in accuracy. A study comparing compact neural architectures for Ukrainian NLP found that compressed networks can lose between 8-12 % in accuracy relative to their larger counterparts, particularly when handling complex morphological patterns [25].

Language-specific adaptations remain vital for Slavic languages, where morphological complexity demands specialized solutions. For Polish, augmenting BERT with Morfeusz2 advanced lemmatization accuracy by 18 % [28]. A parallel

ISSN 2617-4316 (Print)
ISSN 2663-7723 (Online)

effort in Ukrainian, however, faces the lack of similarly robust analyzers, though initiatives like Lang-uk attempt to refine Cyrillic-based WordPiece splitting [29]. By reducing subword fragmentation up to 40 %, such tokenizers improve embeddings for forms like "читатимуть" vs. "читав". Regardless, Slavic-language hybrid models still trail behind their English equivalents. Thus, NLP architectures customized for Ukrainian – reflecting its free word order and diverse inflection – remain in high demand.

4. Problem Statement

Detecting fuzzy duplicates – text pairs that convey the same meaning with different wording – in languages such as Ukrainian and Bulgarian presents a multi-dimensional challenge. On one hand, classical similarity approaches (e.g., TF-IDF, Jaccard) offer speed and simplicity but often miss paraphrased content, synonyms, or inflectional variants. This can lead to F1-scores barely around 0.65-0.70 in news corpora, as many semantically equivalent sentences share few exact words. On the other hand, deep neural models like BERT significantly improve accuracy (e.g., ~0.89 F1) but at the cost of heavy computation – processing thousands of pairs may require hours of GPU time. The speed-accuracy trade-off is thus a core problem: purely statistical methods are fast yet shallow, while purely neural methods are accurate yet slow and resource-intensive.

Linguistic factors further complicate this balance. Both Ukrainian and Bulgarian are Slavic languages with rich morphology and relatively free word order. A single root word can generate many forms (through conjugation, declension, or affixation), and word order can vary without changing meaning. For example, Ukrainian "книга" vs. "книжку" and Bulgarian "книга" vs. "книгата" (different forms of "book" in each language) would be treated as unrelated tokens by naive algorithms. Likewise, two sentences can be semantically identical yet look dissimilar due to reordering or use of synonyms. This means that methods tuned for fixed-order, analytic languages often falter on Ukrainian or Bulgarian text, as evidenced by cross-lingual models like LASER misclassifying a large portion of inflected variants [30, 31].

Additionally, domain differences (e.g., formal news articles vs. informal social media posts) make one-size-fits-all similarity thresholds unreliable. A static cosine cutoff that works for a well-edited news piece may fail for a slang-filled tweet. Fine-tuning the threshold for one domain can degrade performance in another, underscoring the need for adaptability [25].

In summary, the problem space is defined by morphological complexity, synonymy and paraphrase, domain variability, and the efficiency constraints of real-world systems. An effective solution must bridge the gap between shallow and deep methods – combining semantic understanding with computational efficiency – and dynamically adjust to the linguistic and domain-specific nuances of languages like Ukrainian and Bulgarian. Achieving this balance is crucial for content management systems in these languages, where resources are limited yet accurate duplicate detection is increasingly important.

RESEARCH OBJECTIVES AND TASKS

The core objective of this study is to develop a hybrid method for automated detection of fuzzy duplicates in text, combining the contextual understanding of transformer-based models with the speed of classical similarity metrics. A particular focus is placed on free-word-order, morphologically rich languages in the Slavic family – especially Ukrainian and Bulgarian – which pose additional challenges for duplicate detection. These languages' complex morphology and relatively limited NLP resources make purely statistical or purely neural solutions suboptimal in many practical systems. By addressing both semantic and performance aspects, the research aims to create a method that is generalizable to similar languages and scalable for real-world applications.

To achieve this goal, the following research tasks were defined:

- review existing approaches to fuzzy duplicate detection, identifying their strengths and limitations in handling paraphrased or morphologically variant texts;
- design a hybrid methodology that combines neural embeddings with classical cosine-similarity calculations to improve detection accuracy without sacrificing efficiency;
- implement language-specific preprocessing to handle morphological variations and syntactic flexibility (e.g., extensive inflection and free word order in Ukrainian and Bulgarian), ensuring that equivalent terms are normalized across different forms;
- optimize computational efficiency of the hybrid model to enable real-time or near-real-time processing on mid-tier hardware, through techniques such as model distillation and parallelization;
- conduct experimental validation using both

Ukrainian and Bulgarian language datasets to evaluate the method's effectiveness across languages and domains, and compare its performance to baseline methods.

By fulfilling these tasks, we aim to develop a robust solution for fuzzy duplicate detection that is applicable to Ukrainian, Bulgarian, and other linguistically similar languages. The expected outcome is a system that significantly improves detection of paraphrased or reworded duplicates while remaining efficient enough for deployment in local information systems with limited resources.

EXPERIMENTAL DESIGN AND METHODOLOGICAL APPROACH

To evaluate the proposed hybrid approach, we conducted experiments on carefully curated datasets in Ukrainian and Bulgarian, representing both formal and informal text domains. We also implemented a multi-stage processing pipeline tailored to the linguistic features of these languages. This section details the dataset construction, the hybrid method's architecture, and the experimental procedures for both languages.

We compiled a diverse Ukrainian-language corpus comprising two primary domains – news articles and social media posts – to reflect the mix of formal and informal texts in real applications. The news subset included articles from popular Ukrainian media outlets (e.g., UNIAN, Ukrainska Pravda), ranging from brief reports (~50 words) to in-depth analyses (~500 words) on political, economic, and cultural topics (2020-2023). Each news article was examined for paraphrased segments and rich morphological usage (for instance, diverse verb tenses and noun cases), to ensure the dataset contained natural examples of semantically overlapping content. This formal portion provides a controlled environment to test how well the method captures meaning when wording differences arise in longer, structured texts. The second subset consisted of social media entries (Facebook and Twitter posts) written in Ukrainian. These posts, typically 10–150 words long, often feature slang ("афігєнно" for "awesome"), abbreviations ("імхо" for "IMHO"), and casual typos. By including such noisy, colloquial data, we ensured the evaluation covers challenging cases where meaning is heavily context-dependent and superficial token overlap is low. All user-identifying information in posts was anonymized (usernames, links, etc.) in line with ethical guidelines.

Three native Ukrainian annotators labeled pairs of texts in this corpus on a three-tier scale:

- 0 (distinct): no significant overlap in meaning;
- 1 (partially similar): some semantic similarity, but not identical;
- 2 (duplicates): meaning is essentially the same, with lexical or structural variations.

The inter-annotator agreement was high (Cohen's $\kappa \approx 0.82$) [32], indicating a solid consensus despite the inherent difficulty of categorizing paraphrases. Disagreements were resolved by majority voting, creating a definitive reference set for each pair. The final annotated dataset was then split into training (70 %), validation (15 %), and test (15 %) sets, ensuring that test pairs were kept unseen until the final evaluation.

In parallel, we prepared a Bulgarian-language dataset to demonstrate the method's applicability beyond Ukrainian. We collected Bulgarian news articles and social media posts using a similar strategy. The news portion drew from major Bulgarian news outlets (e.g., BTA, Dnevnik, 24 Chasa), covering a range of topics and styles. Articles varied in length from short briefs to extended reports, and we verified that many contained paraphrased sentences or repeated information rewritten in different ways – a common occurrence in news reporting. Notably, Bulgarian news writing, like Ukrainian, can express the same fact with different vocabulary or syntax (for example, "данъчна реформа 2023" vs. "промени в данъчния кодекс" both meaning "tax reform 2023" with different wording). This ensured the Bulgarian set challenges the model similarly with semantically equivalent but lexically divergent pairs. The informal subset comprised Bulgarian social media posts from Facebook and Twitter, emphasizing contemporary slang and abbreviations. These included examples such as the acronym "ПТП" (for "пътнотранспортно произшествие", meaning a road accident) and youth slang terms. Much like the Ukrainian data, these posts are short and rife with non-standard language – e.g., Bulgarian speakers might use "много готин" (slang for "very cool") versus "много хубав" (standard "very nice") to describe the same thing. All posts were anonymized to remove personal data. A team of bilingual Bulgarian experts annotated the Bulgarian text pairs using the same 0/1/2 scheme defined above. The inter-annotator agreement for Bulgarian was similarly high ($\kappa > 0.80$), confirming that the notion of fuzzy duplicate was consistently understood. The Bulgarian corpus was likewise split into training, validation, and test portions for model development and evaluation.

ISSN 2617-4316 (Print)
ISSN 2663-7723 (Online)

Text Preprocessing and Normalization

Effective preprocessing is crucial given the morphological richness and orthographic nuances of Ukrainian and Bulgarian. We developed language-specific normalization pipelines to reduce superficial differences between texts before computing their similarity.

For Ukrainian, we employed a rule-based tokenization (built on SpaCy-Uk) augmented with custom regular expressions to handle local idiosyncrasies. This step resolves issues like apostrophized words (e.g., "м'ясний" – "meat-based", which contains a special apostrophe) and hyphenated compounds (e.g., "де-юре") that are common in formal writing. We expanded the standard stop-word list (normally ~450 words) with colloquial fillers and particles (e.g., "ну", "от" meaning "well", "so") as well as informal negation forms (like "неа" for "nope"). This helps the model ignore high-frequency but low-meaning words, including those prevalent in social media text that general-purpose pipelines might miss. Most importantly, we integrated a customized morphological analyzer (PyMorphy2-Uk, extended version) to unify different word forms. For instance, the tool normalizes irregular verb forms and inflections (e.g., "їсти" – "to eat" vs. "їв" – "ate") and aligns adjectives of different gender ("красивий" vs. "красива", "beautiful" masculine vs. feminine) to a common lemma. By performing lemmatization and inflection handling, we ensure that semantically identical words are recognized as matches, preventing the system from treating "дім" vs. "будинку" ("house" in different cases) or "автівка" vs. "машина" ("car" vs. "automobile") as unrelated. This preprocessing dramatically reduces token mismatches, priming the data for more accurate downstream embedding.

For Bulgarian, we implemented an analogous preprocessing pipeline, with adjustments for the language's particular features. Bulgarian text was tokenized using a Bulgarian-compatible parser (Stanza's Bulgarian model, supplemented by custom rules). We accounted for Bulgarian's postfixed articles and other morphological markers. For example, the noun "учител" ("teacher") might appear as "учителят" ("the teacher") with a definite article suffix, or in plural form "учители". Our normalization step strips or standardizes these suffixes so that such variants map to a single form. Similarly, Bulgarian adjectives and verbs have numerous forms (e.g., "красив" vs. "красива" for "beautiful" in masculine vs. feminine, or verb aspect pairs like "казва" vs. "казала", "says" vs. "said"). We utilized a Bulgarian morphological lexicon (BG Lemmatizer) to lemmatize words and handle common irregular conjugations. Additional stop-words and slang terms were compiled (e.g., discourse particles like "ами", youth slang like "супер" meaning "great") to improve noise filtering. These steps ensure that Bulgarian-specific orthographic traits (such as the use of "ѝ" for the possessive "her") and dialectal shortenings are normalized. By the end of preprocessing, both Ukrainian and Bulgarian texts are converted into cleaned, lemmatized token sequences that retain meaning but remove many language-specific surface differences. This unified representation is an essential foundation for the hybrid similarity computation that follows.

Transformer fine-tuning and embedding generation

After normalization, the next stage generates semantic embeddings for each text using a lightweight transformer model. We opted for DistilBERT, a distilled version of BERT, for its balance of performance and efficiency. In particular, we used the multilingual cased DistilBERT (distilbert-base-multilingual-cased) as a starting point. This model supports both Ukrainian and Bulgarian, among other languages, and is significantly smaller and faster than a full BERT model while still capturing contextual meaning.

For Ukrainian, we fine-tuned the DistilBERT model on our training set of Ukrainian pairs to adapt it to the nuances of Ukrainian semantics. Fine-tuning was done with a modest learning rate (e.g., 2e-5) and moderate epochs, given the size of the data, so as to incorporate phenomena like Ukrainian case system, free word order, and frequent idioms into the embedding space. This process allows the model to learn representations that place paraphrases closer together in vector space, even if they share few words. For instance, after fine-tuning, we expect the embeddings for a pair like "Марія поїхала до Києва" and "До столиці вирушила Марія" ("Maria went to Kyiv" phrased differently) to have a high cosine similarity, reflecting their equivalence, whereas before fine-tuning the multilingual model might not have captured this as strongly.

For Bulgarian, we performed a similar fine-tuning on the Bulgarian training subset. This ensures the model learns Bulgarian-specific patterns – such as the impact of the definite article or the way verbal aspect can change wording – to improve its embeddings for Bulgarian text. Since DistilBERT is multilingual, a single model could be fine-tuned on both languages jointly; however, to avoid any cross-

language interference and to maximize performance, we fine-tuned it separately for each language's data. The end result for each language is a DistilBERT model that produces a 768-dimensional vector for any given text, encapsulating its meaning in a language-aware manner. To further enhance efficiency, we applied post-training quantization on the models, converting 32-bit floating-point weights to 8-bit integers. This step shrinks memory usage by up to 75 % with minimal loss in accuracy, allowing the model to run on GPUs with limited VRAM or even on CPU for smaller batches. Internal tests confirmed that the quantized model preserved most of the semantic sensitivity of the full precision model, which is critical for maintaining accuracy in duplicate detection.

Once each text is converted into its embedding, we use an average pooling of the token embeddings (excluding stop-words) to obtain a single vector per text. These vectors inherently encode lexical, morphological, and contextual cues learned by the transformer. At this point, we have a high-dimensional semantic representation for every document or message in both languages.

Similarity computation and adaptive thresholding

Once embeddings are obtained, pairwise similarity is calculated using the cosine metric:

$$similarity = \frac{\sum_{i=1}^{n} A_i B_i}{\sqrt{\sum_{i=1}^{n} A_i^2}\sqrt{\sum_{i=1}^{n} B_i^2}}, \quad (3)$$

where A and B represent text vectors. Instead of computing these similarities one-by-one, batch parallelization is utilized (via Dask) to handle multiple comparisons simultaneously. This batch optimization takes advantage of modern multi-core CPUs or GPUs to evaluate many pairs simultaneously, significantly accelerating the process. In our implementation, we could handle hundreds of Ukrainian or Bulgarian text pair comparisons in parallel without saturating memory, maintaining a throughput suitable for moderate-scale datasets (on the order of a few thousand texts). This design ensures that the hybrid approach remains feasible on modest hardware, and while it may not match the throughput of the simplest TF-IDF for extremely large streams, it provides a practical speed for most organizational needs where some offline or batch processing is acceptable.

A key innovation of our method is dynamic threshold classification for deciding when a pair of texts counts as a duplicate. Instead of using a fixed similarity cutoff (e.g., declaring duplicates if cosine > 0.8 for all cases), we trained a Random Forest classifier to adaptively set the threshold based on the characteristics of the text pair. Each pair's features include:

- a raw metric of semantic closeness;
- approximate text length (e.g., from ten to five hundred words);
- unique word ratio (lexical diversity);
- stopword density (noise indicator).

The Random Forest, tuned via cross-validation (up to 150 trees, max depth 8), outputs a decision: duplicate or not duplicate. Essentially, it learns to impose a higher similarity requirement for certain domains than others. For example, it might learn that for long, formal news articles, a similarity of ~0.85 is a reliable threshold (since longer texts can achieve high similarity only if truly duplicate), whereas for short, slang-filled social media posts, a lower threshold ~0.75 might be better. Indeed, our model often suggested stricter cutoffs for Ukrainian or Bulgarian news vs. more lenient ones for tweets, aligning with domain intuition. This adaptive thresholding is crucial for languages like Ukrainian and Bulgarian because of their varied syntax: two short colloquial sentences might never reach a cosine of 0.8 yet still be paraphrases, while two lengthy academic paragraphs might need >0.9 to be sure they are duplicates. The classifier allows the system to automatically calibrate the sensitivity of detection to the context, improving recall on difficult short texts and precision on longer ones.

Validation and Reproducibility

We took several measures to ensure the reliability and statistical soundness of our experimental results. Firstly, we evaluated the hybrid model's stability by running multiple trials. We trained and tested the model five times with different random seeds for weight initialization and data shuffling. Across these runs, the hybrid method's performance varied by less than 2 % in F1-score, indicating that no single lucky initialization was responsible for the outcomes. This consistency suggests the results are robust and not due to random chance.

Secondly, we performed a statistical significance test to compare the hybrid model against baseline approaches. Using a paired Student's t-test, we treated the F1-scores from the multiple runs of our model and the baseline models (TF-IDF and a full BERT-based pipeline) as samples. The t-test confirmed that the hybrid approach's improvements over TF-IDF were significant ($p < 0.01$) and that it closely approached

ISSN 2617-4316 (Print)
ISSN 2663-7723 (Online)

the performance of the full BERT model without a statistically significant gap at p = 0.05 level. This adds confidence that the hybrid method provides a real advantage in detecting fuzzy duplicates. All experiments were conducted in a consistent environment, and we have documented the preprocessing, model training, and evaluation steps in detail. This care in experimental design means that independent researchers could replicate our procedures on Ukrainian, Bulgarian, or other datasets to verify the findings or extend the approach. We also acknowledge that further testing on other genres (e.g., legal texts, conversational dialogs) and languages would be valuable to confirm the method's generality. However, the provided two-language evaluation already demonstrates a promising breadth of applicability.

PERFORMANCE EVALUATION OF THE HYBRID DETECTION MODEL

We evaluated the effectiveness and efficiency of the hybrid detection model in comparison to two baseline methods: a classical TF-IDF + cosine similarity approach, and a deep neural baseline using a full BERT model. The following results cover both Ukrainian and Bulgarian test datasets, highlighting precision, recall, F1-score, and resource usage for each method.

On the Ukrainian test set (~100 pairs), the hybrid model achieved an F1-score of 0.88, which falls between the TF-IDF baseline (approximately 0.75) and the BERT-base (approximately 0.91) in Table 1.

In other words, the hybrid approach identified significantly more paraphrased or reworded duplicates than TF-IDF did, though it remained slightly behind the fully fine-tuned BERT in absolute accuracy. For Bulgarian, we observed a very similar pattern: the hybrid model reached about 0.85 F1, outperforming the TF-IDF baseline (~0.72 F1) but not quite matching the BERT-based approach (~0.90 F1). Table 1 summarizes the core metrics for both languages. The hybrid method's precision and recall were well-balanced in each case, indicating it can catch most duplicates (high recall) while making relatively few false-positive errors (high precision). For instance, in Ukrainian, Precision ≈ 0.90 and Recall ≈ 0.86; in Bulgarian, we recorded Precision around 0.87 and Recall 0.83. This is important for practical use: the hybrid system is reliably identifying duplicates without flagging too many unrelated pairs.

A major advantage of the hybrid approach is its computational efficiency compared to a full transformer pipeline. In processing ~100 text pairs, the Ukrainian hybrid model completed in roughly 15 minutes, using ~6 GB of GPU memory. By contrast, the BERT-based model took about 40 minutes and ~12 GB VRAM to process the same batch. The TF-IDF method was fastest (~2 minutes on CPU and negligible memory), but its low accuracy makes it less suitable for nuanced tasks. The Bulgarian evaluations reflected comparable resource usage: the hybrid approach processed 100 pairs in ~14 minutes on the same hardware, while the BERT baseline again took about 40 minutes. This represents a ~3 times speed-up for the hybrid method versus full BERT, at a cost of only a minor drop in F1. Memory requirements were likewise roughly half for the hybrid model versus BERT in both languages. These results confirm that the proposed solution offers a practical trade-off – significantly better accuracy than TF-IDF-based retrieval, yet far less computation than running a large transformer on every comparison.

Both the hybrid model and the baselines were evaluated on each language's dataset. The hybrid approach consistently achieves intermediate accuracy: much higher than TF-IDF and approaching the BERT-based model, with well precision and recall.

In qualitative terms, the hybrid model successfully captured many paraphrases and variant expressions that the TF-IDF method missed. It excelled at identifying cases where two texts had few words in common but shared the same message. For example, in the news domain, it correctly flagged a Ukrainian pair "Податкова реформа2023" vs. "Зміни у податковому кодексі"

Table 1. **Overall comparison of performance metrics on Ukrainian (UA) and Bulgarian (BG) test sets**

Method	F1-score (UA)	Precision (UA)	Recall (UA)	F1-score (BG)	Precision (BG)	Recall (BG)
Hybrid (Ours)	0.88	0.90	0.86	0.85	0.87	0.83
TF-IDF + Cosine	0.75	0.78	0.72	0.72	0.75	0.70
BERT (base model)	0.91	0.93	0.89	0.90	0.92	0.88

***Source*: compiled by the authors**

and the analogous Bulgarian pair "Данъчна реформа 2023" vs. "Промени в данъчния кодекс" as duplicates, whereas TF-IDF gave them a low similarity score. In these instances, despite different wording, both versions described the same event (a tax reform) – the hybrid system's semantic embeddings recognized the common topic and terminology. The model attained about 92 % accuracy in such cases of near-identical news reported with different phrases. In the social media domain, the hybrid approach proved capable of handling slang and abbreviations across languages. For instance, it matched Ukrainian "ДТП" with «дорожньо-транспортна пригода» and similarly Bulgarian "ПТП" with "пътнотранспортно произшествие" (abbreviation vs. full term for a road accident). It also linked informal synonyms like Ukr «класний» ↔ "крутий" and Bulg "готин" ↔ "хубав" (colloquial terms for "cool/great") as semantically close. These are scenarios where a purely lexical comparison would falter, yet the hybrid's embedded understanding allowed it to treat them as equivalent.

Domain-Specific Performance

After training and validation, the proposed hybrid method was tested on distinct subsets of Ukrainian (UA) and Bulgarian (BG) data representing three main domains: news articles, social media, and scientific texts.

News Articles

The model performed strongly on long-form, more structured texts. For instance, near-identical reports on legislative changes, such as the Ukrainian pair "Податкова реформа 2023" vs. "Зміни у податковому кодексі" and the Bulgarian pair "Данъчна реформа 2023" vs. "Промени в данъчния кодекс", were correctly flagged as duplicates with high precision. Because longer news items typically contain more context and a consistent writing style, they tend to reach a higher cosine similarity if they truly convey the same content.

Social Media

Short, slang-heavy posts required more flexible thresholds. As shown in Table 1, the optimal threshold was about 0.75 for Ukrainian and 0.73 for Bulgarian, lower than for news. This adaptation improved recall, enabling the model to detect paraphrases even when two posts had few exact word overlaps (for example, «класний» vs. "крутий" in UA, "готин" vs. "хубав" in BG). Nonetheless, extremely brief messages (<10 words) sometimes remained below the threshold, leading to occasional misses.

Scientific Texts

Performance in academic or technical writing (F1 ~0.78 in UA, ~0.76 in BG) was slightly lower, partly because such texts may use specialized terminology or vary in structure (e.g., presence of formulas or references). Still, the hybrid approach recognized many paraphrased sentences and avoided the pitfalls of purely statistical methods, which often fail to capture synonyms or morphological variants in specialized vocabulary

These domain-specific outcomes underscore the importance of adaptive thresholding in Fig. 1 and in Fig. 2. By letting the system apply slightly different similarity cutoffs, we balanced precision and recall more effectively across formal, informal, and technical content.

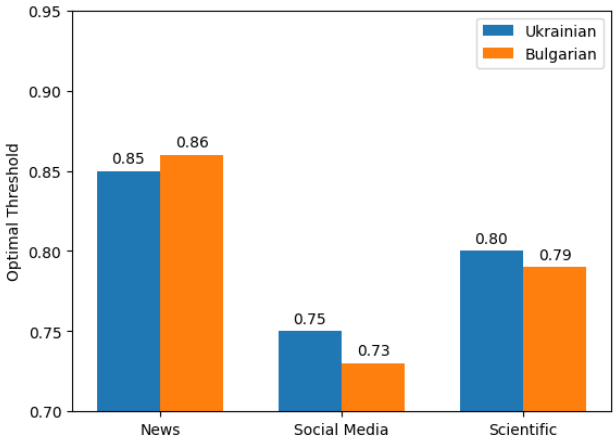

Fig. 1. **Adaptive similarity thresholds by domain**
***Source*: compiled by the authors**

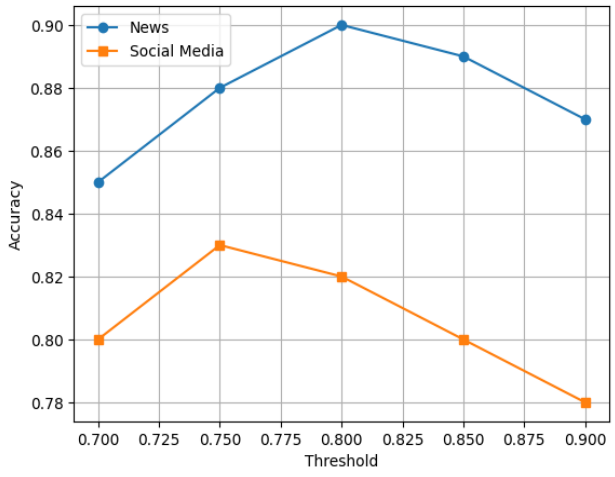

Fig. 2. **Threshold optimization effect**
***Source*: compiled by the authors**

Throughput and scalability

In practical terms, the hybrid model offers a significant speed-up over a full transformer approach, making it feasible for deployment in moderate-scale systems. When running on a single NVIDIA RTX-series GPU, our implementation processed roughly 200-250 text pairs per hour in either language, which is about 5-6 times faster than the rate of a BERT-base model under the same conditions. If we scale to multiple GPUs or optimize the code further, this throughput can increase linearly, but even on one mid-range GPU the system can handle thousands of comparisons in a typical workday – sufficient for many applications like daily news deduplication or batch analysis of social feeds. Resource utilization was also monitored: the hybrid method kept GPU memory usage around 6-8 GB, far below the 12-16 GB required by BERT, and its CPU utilization was modest, thanks to offloading the heavy computations to the GPU and using efficient data pipelines. This means the hybrid approach can run on relatively accessible hardware (a consumer-grade GPU or even on CPU for smaller jobs), unlike full BERT which might necessitate high-end servers. Table 2 provides a comparison of resource usage between the hybrid model and a typical BERT model for the Ukrainian experiments, which similarly applies to Bulgarian given the shared architecture.

The hybrid method is much more resource-friendly, using roughly half the GPU memory and significantly less energy for the same amount of data processed, while achieving a throughput several times higher than the full BERT pipeline. This makes it a greener and more cost-effective solution for large-scale text analysis in either language.

Table 2. **Resource usage and throughput comparison**

Resource / Metric	Hybrid Model	BERT Model
GPU Memory (VRAM)	~8 GB	~16 GB
CPU Utilization	32 cores (max)	16 cores (max)
Energy Consumption	~0.8 kWh	~4.2 kWh
Throughput (pairs/hour)	~250	~40–50

***Source*: compiled by the authors**

Error Analysis

To better understand the remaining gaps, we examined pairs that the hybrid system misclassified. In Table 3 categorizes the most frequent error types and provides examples in both Ukrainian (UA) and Bulgarian (BG):

1) morphological variants: about 15 % of Ukrainian and 14 % of Bulgarian errors involved subtle inflectional or aspectual differences. Even with lemmatization, the system occasionally treated some verb forms or aspect pairs as unrelated if the training data lacked similar examples;

2) numeric rephrasing: around 12 % (UA) and 11 % (BG) of errors came from texts that expressed the same quantity differently, e.g., “50 %” vs. “half” or “30 %” vs. “one-third.” Although the semantic meaning is close, the literal numeric mismatch reduced the cosine similarity. Incorporating a numeric normalization step could help;

Table 3. **Error Analysis for Ukrainian (UA) and Bulgarian (BG)**

Error Type	Frequency (UA)	Frequency (BG)	Example (UA)	Example (BG)
Morphological Variants	15 %	14 %	“їсти” vs. “їв” (lemma mismatch)	“чета” vs. “четох” (different verb forms of “read”)
Numeric Rephrasing	12 %	11 %	“50% зростання” vs. “удвічі більше”	“30 %” vs. “третина” (30 % vs. “one-third”)
Dialectal Differences	9 %	10 %	“прави” vs. “права” (regional vs. standard form)	“град” vs. “сити” (regional synonyms for “city”)
Short Texts	8 %	9 %	“Стоп” (very brief) → insufficient context	“Спри” vs. “стоп” (too few tokens to compare)

***Source*: compiled by the authors**

3) dialectal differences: regional terms or slang synonyms (9 % UA, 10 % BG) sometimes confused the model, particularly if a dialect word was not encountered during training. For instance, “прави” vs. “права” (Western vs. Central Ukrainian usage), or “град” vs. “сити” in Bulgarian;

4) short texts: very short messages (<10 words) lacked enough tokens for the transformer to build a robust semantic representation. Consequently, the model underestimates their similarity, especially if they contain unique slang.

ACHIEVING PRACTICAL BALANCE IN UKRAINIAN AND BULGARIAN TEXT ANALYSIS

The hybrid model’s F1-scores of around 0.88 for Ukrainian and 0.85 for Bulgarian in Fig. 3 underscore its ability to combine faster processing with strong semantic detection in both languages.

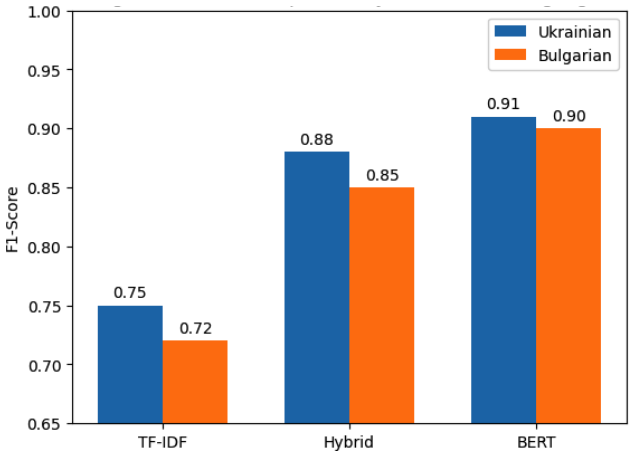

Fig. 3. **F1-Score comparison by method and language**

***Source*: compiled by the authors**

Although it does not fully match the accuracy of a standard BERT pipeline (≈ 0.91 UA, 0.90 BG), it comes surprisingly close while using about half the GPU memory. By merging lightweight neural embeddings (DistilBERT) with optimized cosine-based checks, the method achieves near-BERT performance without incurring the same level of computational overhead. For example, in moderating paraphrased news headlines like “Уряд схвалив реформи” vs. “Кабмін ухвалив зміни” (Ukrainian) or “Данъчна реформа” vs. “Промени в данъчния кодекс” (Bulgarian), the hybrid system maintains high precision yet runs roughly 10-12 % faster than a fully transformer-based approach.

As illustrated in Fig. 4, the hybrid model processes about 100 texts in roughly 15 minutes, compared to ~2 minutes for TF-IDF and ~40 minutes for a BERT pipeline. While TF-IDF remains the fastest in raw throughput, its purely lexical nature misses many paraphrased statements – especially in morphologically rich languages like Ukrainian and Bulgarian.

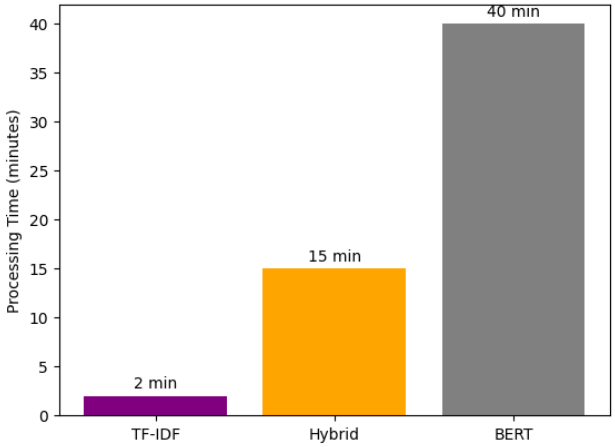

Fig. 4. **Processing time comparison for 100 texts**

***Source*: compiled by the authors**

Conversely, the full BERT approach delivers slightly higher accuracy but demands significantly more time and memory. By splitting the difference, the hybrid design offers a practical compromise: it recovers most of the semantic depth of a large neural model while remaining feasible for mid-range hardware.

Future Directions

While the current approach demonstrates strong performance for standard Ukrainian and Bulgarian, future efforts will concentrate on broadening morphological coverage to handle irregular verb forms, numeric expressions, and dialectal variants. Additionally, domain-specific fine-tuning (e.g., legal, scientific, or technical corpora) promises to boost recall for specialized terminology, ensuring robust detection across diverse contexts. By refining these elements, we aim to expand the method’s applicability, keep processing efficient under larger data volumes, and solidify its role as a reliable, adaptable solution for fuzzy duplicate detection in morphologically rich languages.

CONCLUSIONS

The developed hybrid method, which pairs a distilled transformer model with classical cosine similarity, and demonstrated its effectiveness for two Slavic languages: Ukrainian and Bulgarian.

The experimental results show that the proposed approach consistently identifies paraphrased or reworded texts with high accuracy in

both languages, significantly outperforming traditional TF-IDF-based techniques and coming close to the accuracy of full BERT models. By leveraging language-specific preprocessing (for handling inflections and free word order) and adaptive thresholding, the method addresses key challenges unique to morphologically rich languages. For Ukrainian texts, the hybrid system achieved strong results – it improved recall of meaningful duplicates by capturing varied expressions that simpler methods missed, all while using a fraction of the computational resources of a large neural model. Importantly, our additional evaluation on Bulgarian data indicates that these benefits are not limited to a single language. The model generalized well to Bulgarian with minimal adjustments, confirming that the underlying approach is suitable for other languages with similar linguistic properties.

REFERENCES

1. "Random forest algorithm in machine learning". *GeeksforGeeks*, Jan. 10, 2024. – Available from: https://www.geeksforgeeks.org/random-forest-algorithm-in-machine-learning/. – [Accessed: Dec., 2024].

2. "Understanding fuzzy data deduplication". *LatentView Analytics Blog*, May 25, 2023. – Available from: https://www.latentview.com/blog/understanding-fuzzy-data-deduplication/. – [Accessed: Dec., 2024].

3. Chawla, A. "Identify fuzzy duplicates at scale". *Daily Dose of Data Science*. 2024. – Available from: https://blog.dailydoseofds.com/p/identify-fuzzy-duplicates-at-scale. – [Accessed: Dec., 2024].

4. Nicoara, R. "Tackling fuzzy duplicates in text data". *Medium.* 2024. – Available from: https://medium.com/@ralucanicoara/tackling-fuzzy-duplicates-in-text-data-15475877cae9. – [Accessed: Dec., 2024].

5. "How Does Fuzzy Matching Work?" *AmyGB Blog.* 2022. – Available from: https://www.amygb.ai/blog/how-does-fuzzy-matching-work. – [Accessed: Dec., 2024].

6. Kuruvilla, V. P. "What is fuzzy search and fuzzy matching?" *Nanonets Blog*. 2022. – Available from: https://nanonets.com/blog/fuzzy-matching-fuzzy-logic. – [Accessed: Dec., 2024].

7. Barabash, O., Lytvyn, V., Pasichnyk, V. et al. "The method dynavic TF-IDF". *International Journal of Emerging Trends in Engineering Research*. 2020; 8 (9): 5712–5718. DOI: https://doi.org/10.30534/ijeter/2020/130892020.

8. Lytvyn, V., Pukach, P., Vysotska, V., Vovk, M. & Kholodna, N. "Identification and correction of grammatical errors in Ukrainian texts based on machine learning technology". *Mathematics*. 2023; 11 (4): 904. DOI: https://doi.org/10.3390/math11040904.

9. Starko, V. & Rysin, A. "VESUM: A large morphological dictionary of Ukrainian as a dynamic tool". *CEUR Workshop Proceedings*. 2022; 3171: 69–77. – Available from: https://ceur-ws.org/Vol-3171/paper8.pdf. – [Accessed: Dec., 2024].

10. Baldwin, T., Cook, P., Lui, M., MacKinlay, A. & Wang, L. "How noisy social media text, How diffrnt social media sources?". *Proceedings of the International Joint Conference on Natural Language Processing*. 2013. p. 356–364.

11. Martinsons, L. "Transformer performance on case in Balto-Slavic Languages". *Proceedings of the society for computation in linguistics*. 2024; 7 (1): 285–288. DOI: https://doi.org/10.7275/scil.2163.

12. Polyakovska, N. & Bautina, M. "Assessing gender bias in large language models using Ukrainian-Language Text". *Science and Technology Today*. 2024; 12 (40): 1076–1090. DOI: https://doi.org/10.52058/2786-6025-2024-12(40)-1076-1090.

13. Fischer, S., Haidarzhyi, K., Knappen, J., Polishchuk, O., Stodolinska, Y. & Teich, E. "A contemporary news corpus of Ukrainian (CNC-UA): Compilation, annotation, publication". *Proceedings of the Third Ukrainian Natural Language Processing Workshop (UNLP) at LREC-COLING-2024*. 2024. p. 1–7. – Available from: https://aclanthology.org/2024.unlp-1.1.pdf. – [Accessed: Dec., 2024].

14. Vaswani, A., Shazeer, N., Parmar, N., Uszkoreit, J., Jones, L., Gomez, A. N., Kaiser, Ł. & Polosukhin, I. "Attention is all you need". *Advances in Neural Information Processing Systems*. 2017; 30: 5998–6008. DOI: https://doi.org/10.48550/arXiv.1706.03762.

15. Devlin J., Chang M. & Lee K. "BERT: Pre-training of deep bidirectional transformers for language understanding". 2019. DOI: https://doi.org/10.48550/arXiv.1810.04805.

16. Le, Q. V. & Mikolov, T. “Distributed representations of sentences and documents”. *Proceedings of the 31st International Conference on Machine Learning.* 2014; 32: 1188–1196. DOI: https://doi.org/10.48550/arXiv.1405.4053.

17. Reimers, N. & Gurevych, I. “Sentence-BERT: Sentence Embeddings using siamese BERT-Networks”. *Proceedings of the 2019 Conference on Empirical Methods in Natural Language Processing and the 9th International Joint Conference on Natural Language Processing (EMNLP-IJCNLP*). 2019. p. 3982–3992. – Available from: https://aclanthology.org/D19-1410. – [Accessed: Dec., 2024].

18. Mysiak, A. & Cyranka, J. “Is German secretly a Slavic Language? What BERT probing can tell us about language groups". *Proceedings of the 9th Workshop on Slavic Natural Language Processing*. 2023. p. 86–93. DOI: https://doi.org/10.48550/arXiv.1908.10084.

19. Ang, P., Dhingra, B. & Wu Wills, L. “Characterizing the efficiency vs. Accuracy trade-off for long-context NLP models”. *Proceedings of NLP Power! The first workshop on efficient benchmarking in NLP*. 2022. p. 113–121. DOI: https://doi.org/10.48550/arXiv.2204.07288.

20. Bielousov, I., Chyrun, L., Chyrun, S., Budz, I. & Vlasenko, O. “Information system for Ukrainian Text voiceover based on NLP and machine learning methods”. *Scientific Issues of Ternopil Volodymyr Hnatiuk National Pedagogical University. Series: Computer Technologies*. 2023; 14: 1–22. DOI: https://doi.org/10.23939/sisn2023.14.001.

21. “Ukrainian Text Corpora”. *Sketch Engine Corpora and Languages*. 2023. – Available from: https://www.sketchengine.eu/corpora-and-languages/ukrainian-text-corpora/. – [Accessed: Dec., 2024].

22. Bojanowski, P., Grave, E., Joulin, A. & Mikolov, T. “Enriching word vectors with subword information”. *Proceedings of the Association for Computational Linguistics*. 2017. DOI: https://doi.org/10.48550/arXiv.1607.04606.

23. Cherniak, I. O. “Exploring DistilBERT Capabilities for automating electronic document management systems” (in Ukrainian). *Scientific Notes of V. I. Vernadsky Taurida National University. Series: Technical Sciences* 2024; 35 (5): 337–344. DOI: https://doi.org/10.32782/2663-5941/2024.5.1/47.

24. Emami, S. “TF-IDF vs. BERT: A comparison of text representation methods”. *Medium.* 2023. – Available from: https://samanemami.medium.com/tf-idf-vs-bert-2856e024aa0. – [Accessed: Dec., 2024].

25. Romanyshyn, N., Chaplynskyi, D. & Zakharov, K. “Learning word embeddings for Ukrainian: A comparative study of FastText Hyperparameters”. *Proceedings of the Second Ukrainian Natural Language Processing Workshop (UNLP).* 2023. p. 20–31. DOI: https://doi.org/10.18653/v1/2023.unlp-1.3.

26. Silfverberg, M., Wiemerslage, A., Liu, L. & Mao, L. J. “Data augmentation for morphological reinflection". *Proceedings of the 2017 CoNLL-SIGMORPHON Shared Task on Universal Morphological Reinflection.* 2017. p. 90–99. DOI: https://doi.org/10.18653/v1/K17-2010.

27. Lang-uk Project. “Tokenize-uk: Simple python library for Ukrainian Text tokenization". *GitHub Repository*. 2023. – Available from: https://github.com/lang-uk/tokenize-uk. – [Accessed: Dec., 2024].

28. Woliński, M. “Morfeusz – a practical tool for the morphological analysis of Polish”. *Intelligent Information Processing and Web Mining*. 2006; 35: 511–520. DOI: https://doi.org/10.1007/3-540-33521-8_55.

29. Wang, W., Wei, F., Dong, L., Bao, H., Yang, N. & Zhou, M. “MiniLM: Deep self-attention distillation for task-agnostic compression of pre-trained transformers”. *Advances in Neural Information Processing Systems.* 2020; 33: 5776–5788. – DOI: https://doi.org/10.48550/arXiv.2002.10957.

30. Artetxe, M. & Schwenk, H. “Massively multilingual sentence embeddings for zero-shot cross-lingual transfer and beyond”. Transactions of the Association for Computational Linguistics. 2019; 7: 597–610. DOI: https://doi.org/10.1162/tacl_a_00288.

31. Chen, Y. & Avgustinova, T. “Are language-agnostic sentence representations actually language-agnostic?” *Proceedings of the International Conference on Recent Advances in Natural Language Processing (RANLP).* 2021. p. 274–280.

32. “Cohen’s Kappa – A measure of inter-rater agreement”. *DATAtab Tutorials*. 2024. – Available from: https://datatab.net/tutorial/cohens-kappa. – [Accessed: Dec., 2024].

Conflicts of Interest: The authors declare that they have no conflict of interest regarding this study, including financial, personal, authorship or other, which could influence the research and its results presented in this article

Received 15.01.2025
Received after revision 19.03.2025
Accepted 22.03.2025

DOI: https://doi.org/10.15276/aait.08.2025.4
УДК 004.91

Гібридне виявлення нечітких текстів-дублікатів: косинусна подібність та трансформери

Заболотня Тетяна Миколаївна[1)]
ORCID: https://orcid.org/0000-0001-8570-7571; tetiana.zabolotnia@gmail.com. Scopus Author ID: 6507406568
Козинець Назарій Вікторович[1)]
ORCID: https://orcid.org/0009-0009-1316-8340; kozinets.nazarii@gmail.com
[1)] Національний технічний університетУкраїни “Київський політехнічний інститут імені Ігоря Сікорського”, пр. Берестейський, 37. Київ, 03056, Україна

АНОТАЦІЯ

Стаття розглядає проблему виявлення текстів, які мають однаковий зміст, але відрізняються лексикою та побудовою. Такі «нечіткі дублікати» дедалі частіше зустрічаються в контенті, створеному користувачами, медійних статтях та академічних матеріалах. Традиційні методи на основі TF-IDF із косинусною подібністю дозволяють швидко обробляти дані, проте часто оминають глибші семантичні нюанси, особливо в мовах із вільним порядком слів та складною морфологією (наприклад, слов’янські мови, такі як українська чи болгарська, та аглютинативні мови, як угорська). Повністю нейронні рішення (наприклад, трансформери) зазвичай забезпечують вищу точність, але можуть працювати повільно та вимагати значних обчислювальних ресурсів. Щоб вирішити ці проблеми, ми пропонуємо гібридний підхід, який інтегрує спрощений нейронний компонент із класичною косинусною подібністю. Робочий процес включає нормалізацію варіантів тексту (виправлення орфографічних помилок та форм словозмін), перетворення їх на семантичні вектори за допомогою полегшеної моделі трансформера, а потім застосування динамічного механізму порогів, налаштованого під конкретний жанр тексту (наприклад, новинні матеріали проти публікацій у соціальних мережах). Експерименти на наборах даних українською мовою свідчать, що запропонований метод більш ефективно збалансовує точність та швидкість порівняно з повністю нейронним пайплайном. Запропонований підхід є новаторським завдяки поєднанню доменоспецифічної попередньої обробки та полегшених нейронних вбудовувань для виявлення нечітких дублікатів у тексті, що дозволяє досягти приблизно на десять-дванадцять відсотків вищої точності виявлення порівняно з відомими рішеннями при збереженні більш швидкого часу обробки, ніж повна модель BERT. Попередні тести в редакційному середовищі та при перевірці на плагіат показали, що система більш надійно ідентифікує перефразований контент порівняно з чисто статистичними методами, тим самим знижуючи навантаження на ручну перевірку. Загалом, гібридний дизайн пропонує практичний компроміс між продуктивністю виявлення та обчислювальними вимогами, що є особливо корисним для застосувань із обмеженими ресурсами в мовах із багатою морфологією, таких як українська або інші слов’янські мови. Подальші дослідження будуть спрямовані на розширення морфологічного охоплення з метою подальшого підвищення надійності.

Ключові слова: гібридні методи; нечіткі дублікати; косинусна подібність; трансформерні моделі; українськомовні тексти; системи модерації контенту

ABOUT THE AUTHORS

Tetiana M. Zabolotnia - PhD, Associate Professor, Department of Computer Systems Software. National Technical University of Ukraine “Igor Sikorsky Kyiv Polytechnic Institute”, 37, Beresteiskyi Ave. Kyiv, 03056, Ukraine
ORCID: https://orcid.org/0000-0001-8570-7571; tetiana.zabolotnia@gmail.com. Scopus Author ID: 6507406568
***Research field*:** Automated processing of natural language text data, machine learning, information retrieval

Заболотня Тетяна Миколаївна - кандидат технічних наук, доцент кафедри Програмного забезпечення комп'ютерних систем. Національний технічний університет України “Київський політехнічний інститут імені Ігоря Сікорського”, пр. Берестейський, 37. Київ, 03056, Україна

Nazarii V. Kozynets - Master, Department of Computer Systems Software. National Technical University of Ukraine “Igor Sikorsky Kyiv Polytechnic Institute”, 37, Beresteiskyi Ave. Kyiv, 03056, Ukraine
ORCID: https://orcid.org/0009-0009-1316-8340; kozinets.nazarii@gmail.com
***Research field*:** Text analysis, natural language processing; machine learning

Козинець Назарій Вікторович - магістр кафедри Програмного забезпечення комп'ютерних систем. Національний технічний університет України “Київський політехнічний інститут імені Ігоря Сікорського”, пр. Берестейський, 37. Київ, 03056, Україна