Issue 17, 2025

https://doi.org/10.23939/sisn2025.17.044

УДК 004.9

INTELLECTUAL ANALYSIS OF TEXTUAL DATA IN SOCIAL NETWORKS USING BERT AND XGBOOST

Taras Batiuk[1], Dmytro Dosyn[2]

[1, 2] Lviv Polytechnic National University,
Information Systems and Networks Department, Lviv, Ukraine
[1] E-mail: taras.m.batiuk@lpnu.ua, ORCID: 0000-0001-5797-594X
[2] E-mail: dmytro.h.dosyn@lpnu.ua, ORCID: 0000-0003-4040-4467

This article presents a comprehensive approach to sentiment analysis in social networks by leveraging modern text processing methods and machine learning algorithms. The primary focus is the integration of the Sentence-BERT model for text vectorization and XGBoost for sentiment classification. Using the Sentiment140 dataset, an extensive study of text messages labeled with sentiment annotations was conducted. The Sentence-BERT model enables the generation of high-quality vector representations of textual data, preserving both lexical and contextual relationships between words. This contributes to a more accurate semantic understanding of messages, thereby enhancing classification performance. The results of the study demonstrate the high efficacy of the proposed model, achieving an overall classification accuracy of 90 %. The ROC curve (AUC) value of 0.88 further confirms the model's capability to distinguish between sentiment classes effectively. The Precision-Recall curve analysis highlights a strong balance between precision and recall, which is particularly crucial for handling imbalanced datasets. Additionally, calibration curves indicate a high degree of consistency between predicted probabilities and actual outcomes, while the cosine similarity matrix validates the model's ability to capture semantic proximity between texts. Beyond classification, the study also examines the F1-score at various threshold levels, enabling the identification of the optimal operational range for the model. The cumulative gain chart illustrates the progressive improvement in classification performance, emphasizing the model's stability when processing large-scale textual data. The proposed approach serves as a versatile tool for sentiment analysis, text clustering, and trend identification in social networks. The findings of this study have practical implications in fields such as marketing, public opinion analysis, automated content moderation, and social trend prediction.

Keywords – Sentence-BERT, XGBoost, text vectorization, Transformers, cosine similarity.

Problem Statement

In the modern digital era, the analysis of textual and numerical data is gaining increasing importance, particularly in the context of the rapid expansion of social media. Social platforms such as Twitter generate vast amounts of textual information, serving as a valuable resource for studying sentiment, behavioral trends, and user interactions. Given the widespread use of social networks for discussing diverse topics, this data holds significant potential for applications in business, marketing, social research, and even politics. Sentiment analysis in social networks not only provides valuable insights for businesses but also aids in understanding public opinion and identifying key societal discussions. Furthermore, analyzing user sentiments enables the prediction of potential behavioral changes, further highlighting the relevance of such studies.

The analysis of textual data presents challenges due to its unstructured nature, as well as linguistic and stylistic variability. Twitter messages (tweets), as a specific example of textual data, are constrained by character limits, adding complexity to their processing. To effectively handle such data, modern vectorization methods are required to transform text into numerical representations suitable for machine learning algorithms. Among contemporary approaches, BERT (Bidirectional Encoder Representations from Transformers) stands out as a method that captures both lexical and contextual nuances, ensuring high-quality text feature representation. Beyond text vectorization, text classification is a crucial aspect of sentiment analysis. In this context, advanced machine learning algorithms such as XGBoost are employed, offering high performance when processing structured data. XGBoost is a powerful tool for handling vectorized text while also incorporating numerical and categorical features. Due to its optimized approach, this algorithm achieves high accuracy even on complex datasets.

Another critical area of research is the assessment of text similarity. The application of Sentence Embedding Similarity, based on cosine similarity, enables the evaluation of semantic relationships between texts, unlocking new possibilities for data analysis. This technique proves effective in detecting semantically related messages and extracting dominant topics discussed across various texts.

The objective of this study is to develop an integrated approach to sentiment analysis in social networks, leveraging advanced text processing techniques and machine learning algorithms. The research utilizes the Sentiment140 dataset, which contains messages annotated with sentiment labels (positive, negative). The primary aim is to examine the influence of specific vocabulary on classification outcomes and to develop a model capable of accurately identifying user sentiments from textual data. Special attention will be given to the analysis of lexical patterns, which can significantly impact sentiment classification, as well as the integration of diverse approaches to textual data analysis.

The object of this study is textual data, represented as short messages (tweets) annotated with sentiment labels. A key characteristic of this data is its brevity, which complicates analysis and necessitates the use of advanced text processing algorithms to capture relationships between words. Additionally, the presence of informal language, abbreviations, and hashtags in tweets introduces challenges for traditional text analysis methods, further underscoring the relevance of modern approaches. The subject of this study encompasses text analysis methods, including BERT for text vectorization. This approach captures the contextual nuances of text, enabling precise representation of lexical and semantic relationships between words. By leveraging BERT, high-quality vector representations suitable for subsequent machine learning tasks can be generated. Sentence Embedding Similarity is also employed to analyze textual similarity, facilitating the identification of related messages, content-based classification, and semantic proximity assessment. This technique is particularly useful for clustering or filtering data.

For sentiment classification, the study utilizes the XGBoost algorithm. Designed for tabular data processing, XGBoost offers high classification accuracy and efficiently handles both numerical and categorical features. It also benefits from fast training and hyper-parameter optimization, making it particularly advantageous for large datasets. During the study, text features extracted using BERT will be integrated with additional features to construct a unified vector representation of the data. This integration enables the development of a comprehensive sentiment classification model that considers both lexical features and contextual usage. Furthermore, the study examines the impact of specific vocabulary on classification accuracy, identifying keywords that significantly influence sentiment determination. A detailed analysis will be conducted on the correlation between particular keywords or phrases and positive or negative sentiment, which is crucial for enhancing model precision. This examination will provide insights into the model's performance, highlighting strengths and areas for improvement. The study results will be presented through graphical visualizations, aiding in the identification of key data patterns and enhancing the interpretability of conclusions for practical applications. This includes an analysis of feature importance within the model and an investigation of misclassifications, which may indicate areas where methodological refinements are required. Thus, this work combines modern text processing methods, machine learning algorithms, and model evaluation metrics, which allows not only to increase the accuracy of sentiment analysis in social networks, but also to create a platform for further research in this area.

Analysis of recent research and publications

P. Aggarwal (2024) explores the use of BERT and SVM for cyberbullying detection and classification on social media. The study highlights the increasing prevalence of cyberbullying and the need for effective automated detection methods. The authors leverage BERT's deep contextual understanding of language to extract meaningful representations of text, which are then classified using SVM, a robust machine learning algorithm known for its effectiveness in text classification tasks. The study evaluates the model's performance using various metrics, demonstrating high accuracy and reliability in detecting and categorizing cyberbullying content.

D. A. Al-Qudah (2025) presents a new approach for predicting sentiment polarity in restaurant reviews using ordinal regression and an evolutionary-enhanced XGBoost model. The study addresses the challenge of accurately classifying review sentiments, which are often nuanced and contain varying degrees of positivity and negativity. The authors propose an ordinal regression framework instead of traditional classification methods to better capture the ordered nature of sentiment categories. The model is trained and evaluated on a large dataset of restaurant reviews, demonstrating superior accuracy compared to baseline machine learning models.

A. I. Atmaja (2024) investigates the impact of labeling strategies and class-balancing techniques on sentiment analysis for the game Clash of Champions using LSTM and BERT. The study addresses the challenge of imbalanced datasets, which can negatively affect model performance in sentiment classification tasks. The authors compare different data labeling approaches and implement various balancing techniques, such as oversampling, undersampling, and synthetic data generation, to improve sentiment prediction accuracy. While BERT outperforms LSTM in terms of accuracy and robustness, LSTM remains a viable option with lower computational requirements.

K. Aziz (2024) propose a hybrid approach for aspect-based sentiment analysis (ABSA) by integrating BERT with multi-layered Graph CNN. Their study aims to enhance the understanding of sentiment at a fine-grained level, where opinions are analyzed in relation to specific aspects within a text rather than the overall sentiment. The authors leverage BERT's deep contextual embeddings to extract meaningful word representations and then employ multi-layered GCNs to capture semantic relationships between different aspects in a sentence. The results highlight improvements in aspect detection, sentiment classification accuracy, and contextual understanding.

T. Batiuk (2023) presents an intellectual system for clustering social network users based on sentiment analysis of messages. Their study focuses on developing an automated approach to grouping users according to the emotional tone of their online interactions. The system utilizes machine learning and natural language processing (NLP) techniques to analyze text data from social networks, identifying sentiment patterns and similarities among users. T. Batiuk (2024) extends research to develop a decision-making support system for analyzing Twitter publications. This system aims to assist in extracting insights from large volumes of tweets, particularly for sentiment-based decision-making in various fields such as political analysis, brand monitoring, and public opinion research. The proposed system integrates advanced sentiment analysis methods, utilizing machine learning models to provide a comprehensive assessment of Twitter content.

L. He (2024) introduces an enhanced Twitter sentiment analysis model that integrates RoBERTa and BERT within a dual joint classifier framework. The study aims to improve sentiment classification accuracy by leveraging the strengths of both transformer architectures. The proposed dual joint classifier combines BERT's strong contextual understanding with RoBERTa's improved training strategy, which includes dynamic masking and additional pretraining on large datasets. This hybrid approach helps capture nuanced sentiment expressions, especially in complex or ambiguous tweets. The study evaluates the model using benchmark Twitter sentiment datasets, demonstrating superior performance over standalone BERT, RoBERTa, and traditional machine learning models. E. Ivokhin (2022) explores the restructuring of the "State–Probability of Choice" model by utilizing products of stochastic rectangular matrices. Their study focuses on enhancing decision-making models that rely on probabilistic choices, particularly in cybernetics and systems analysis. The authors propose a mathematical framework that refines existing state-probability models, which allows for a more accurate representation of dynamic decision processes.

The research examines how these matrix transformations influence transition probabilities and decision outcomes, making the model more adaptable to complex, real-world scenarios. A. Khan (2025) presents a new approach for sentiment analysis of emoji-fused reviews by combining machine learning techniques and BERT. Their study addresses the challenge of analyzing mixed textual and emoji data, which has become increasingly common in online reviews. By integrating both modalities, the model aims to provide a more comprehensive understanding of sentiment. Experiments on a large dataset of emoji-enhanced reviews show that the approach significantly improves classification accuracy, particularly in identifying subtle sentiment shifts influenced by emojis.

Najeem Olawale Adelakun (2024) focuses on sentiment analysis of financial news using the BERT model. The study explores how sentiment analysis can be applied to financial news articles to predict market trends and provide insights into investor sentiment. The authors use BERT, a powerful deep learning model known for its ability to understand contextual relationships in text, to analyze the sentiment of financial news. They evaluate the model's performance on a dataset of financial news articles to identify positive and negative sentiments. The study shows that BERT is highly effective at capturing complex linguistic structures and improving the accuracy of sentiment classification in the context of finance.

However, several open questions remain regarding the further refinement of these methodologies. Future research should focus on optimizing model training processes for large-scale datasets, considering linguistic and cultural variations in textual data, and developing advanced interactive visualizations to facilitate a more in-depth analysis of user behavior patterns. Additionally, a critical research direction involves enhancing model explainability to provide better insights into decision-making processes, thereby increasing trust in automated sentiment analysis systems.

Formulation of article objectives

This study aims to analyze sentiment in social networks using modern text processing techniques and deep learning methods. The Kaggle dataset Sentiment140 is utilized, which contains messages labeled with sentiment categories (negative, positive). The primary objective of this research is to develop a model capable of classifying the sentiment of messages and estimating the semantic similarity between texts using Sentence-BERT and Sentence Embedding Similarity.

A key methodological approach in this study is the estimation of semantic similarity between texts based on Sentence Embedding Similarity. This technique enables the analysis of textual similarity, content-based grouping, and identification of key topics (Ogunleye, 2024). It facilitates document retrieval, text classification, clustering, and duplicate detection, where A_i and B_i represent the components of vectors A and B, respectively, and n denotes the dimensionality of these vectors. Cosine similarity, which is used to measure the relationship between two vectors, ranges from -1 to 1.

A value of 1 indicates that the vectors are perfectly aligned, 0 signifies that they are orthogonal (perpendicular), and -1 means they point in opposite directions. In the context of text similarity calculations, the values typically range from 0 to 1, as only non-negative vectors are considered. Cosine similarity, as expressed in Formula 1, is widely applied in text clustering, duplicate detection, and the analysis of semantic relationships.

$$\cos(\theta) = \frac{\sum_{i=1}^{n} A_i B_i}{\sqrt{\sum_{i=1}^{n} A_i^2} \times \sqrt{\sum_{i=1}^{n} B_i^2}}, \quad (1)$$

Transformer-based models, particularly Sentence-BERT (S-BERT), have become essential tools for processing textual data. S-BERT enables the efficient conversion of text into numerical representations while preserving the semantic context of words within sentences. This facilitates the application of text data in machine learning algorithms for classification, clustering, and various other text analysis tasks. S-BERT is an optimized modification of BERT designed specifically for computing semantic similarity between texts (Oletsky, 2024). The primary objective of S-BERT, as expressed in Formula 2, where T represents the input text, f_{BERT} denotes the S-BERT transformer function, and v is the resulting fixed-length vector, is to generate compact vector representations that accurately capture the semantic content of texts.

S-BERT incorporates an additional aggregation layer (e.g., mean or max pooling) to produce a single vector representation at the document level. This design enables rapid and efficient computation of semantic similarity between texts.

$$v = f_{BERT}(T), \quad (2)$$

S-BERT significantly enhances the processing of textual data by generating vector representations of sentences that preserve semantic relationships. Its primary advantage lies in its ability to compute semantic similarity between texts, making it particularly useful for sentiment classification, information retrieval, text clustering, and machine translation. K.I. Roumeliotis (2025) noted in his work that the architecture of Sentence-BERT consists of two key components: pre-trained transformer models, such as BERT or DistilBERT, which enable efficient text processing while capturing both syntactic and semantic features, and a pooling operation, which applies an aggregation function, such as mean pooling or max pooling, after the text passes through the transformer layer. This operation converts the original representations into a single fixed-size vector for the entire sentence. A distinguishing feature of S-BERT is its ability to handle pairwise text comparison tasks. For instance, in semantic similarity detection or sentiment classification, S-BERT generates vector representations for each text, after which the similarity between these vectors is computed using metrics such as cosine similarity. This approach enables the model to assess not only the presence of identical words but also the contextual relationships between them. In sentiment classification tasks, S-BERT provides an effective text representation that allows machine learning algorithms, such as XGBoost, to leverage these vector features for highly accurate classification (Setiawan, 2024). The diagram of the S-BERT model is presented in Fig. 1.

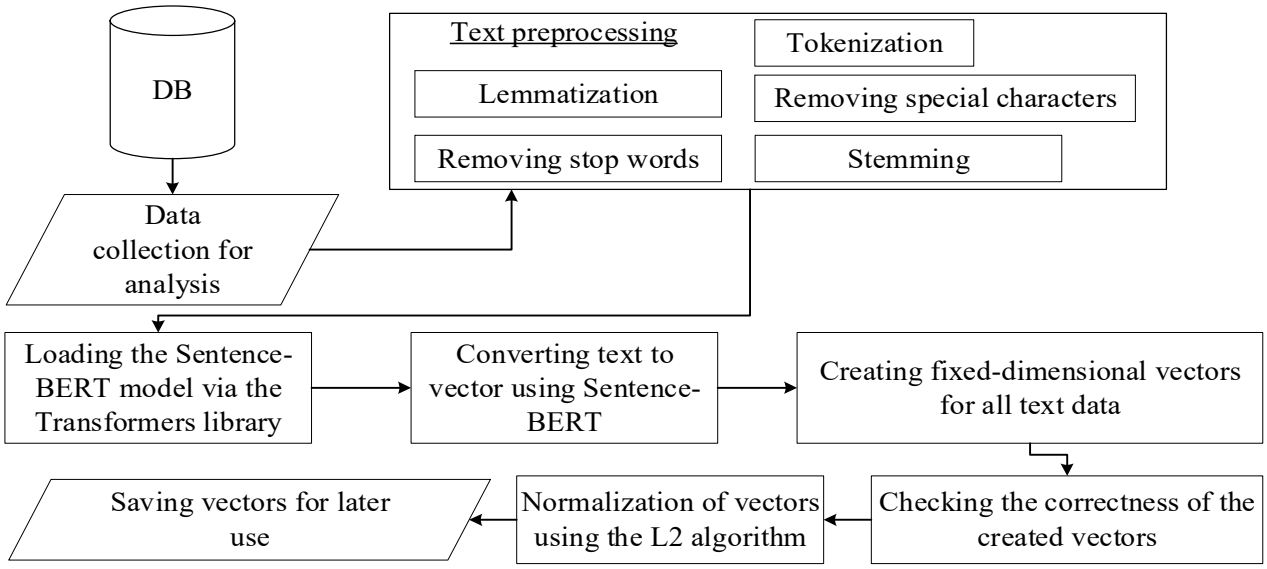

Fig. 1. Diagram of the S-BERT model

While BERT was designed for contextual word embeddings—that is, understanding the meaning of a word based on its surrounding context—Sentence-BERT modifies this approach to generate embeddings for entire sentences. This adaptation is crucial for tasks that require understanding and interpreting the semantic meaning of whole sentences, especially when the sentences are short or ambiguous, which is common in informal social media text. Sentence-BERT operates using a transformer-based architecture, which is composed of multiple layers of attention mechanisms. These layers enable the model to weigh the importance of different words in relation to each other and understand the relationships between them. Transformers excel in capturing long-range dependencies in text, which allows Sentence-BERT to efficiently model the intricate semantic relationships between words in a sentence. This is especially important in complex sentence structures or texts with multiple possible interpretations.

Once the tokens pass through the BERT-based network, the next step in the process is the generation of a sentence embedding. A sentence embedding is a dense vector representation that captures the semantic meaning of the entire sentence. In contrast to traditional word embeddings, which represent the meanings of individual words in isolation, a sentence embedding aggregates the meanings of all words in a sentence.

This means that Sentence-BERT considers how words interact with each other within a sentence, allowing it to understand the overall sentiment or intent expressed by the sentence as a whole. A dense vector refers to a vector of numbers, typically with values in real numbers, where each dimension of the vector captures a different aspect of the sentence's meaning. For Sentence-BERT, the vector representation typically ranges from 256 to 768 dimensions, depending on the specific configuration of the model. Each dimension in the vector is a feature that encapsulates a particular characteristic of the sentence's meaning, such as its tone, sentiment, or the relationships between the words. The exact size of the vector determines how much detail the model captures from the sentence. The higher the number of dimensions, the more information is represented, though this may come with an increased computational cost.

Unlike traditional word embeddings like Word2Vec or GloVe, which map individual words to vectors, Sentence-BERT generates a vector that captures the meaning of an entire sentence. Word embeddings are typically trained on large corpora and represent words as points in a vector space, where semantically similar words are located closer together. However, these word embeddings are unable to capture the relationships between words within a sentence, which is vital for understanding the true meaning of the sentence. The sentence embedding generated by Sentence-BERT is a highly context-aware representation. The architecture of BERT allows for bidirectional attention, meaning the model can evaluate both the previous and following words in a sentence. This bidirectional attention is what enables Sentence-BERT to fully understand the context of the sentence, including any ambiguities or subtle meanings that might arise from the interaction of words. In the case of ambiguous words, such as "bank," Sentence-BERT can distinguish between the different meanings of the word based on the context in which it appears.

The approach of using sentence embeddings is especially advantageous when dealing with social media text, such as messages from platforms like Twitter, where sentences can be short, fragmented, and filled with informal language, slang, abbreviations, and even emojis. These features make social media text challenging to analyze using traditional word-based models, which may struggle to account for the nuances and informal nature of the language. Additionally, Sentence-BERT's ability to understand the context of ambiguous terms, slang, and emotional expressions is invaluable in a social media environment. It can accurately interpret sentences even when the language is non-standard or when words are used in ways that deviate from traditional grammatical rules.

The process of generating the sentence embedding begins after the text has been tokenized. Each word (or subword) is represented as a vector, which is passed through the BERT network. The transformer architecture of BERT processes each token with attention layers, capturing relationships between tokens at various layers. Finally, the output embedding is generated for the entire sentence. This embedding is a single, fixed-length vector that encodes the full meaning of the sentence, making it suitable for tasks like sentiment analysis, where the goal is to classify the overall sentiment of a message or statement. By leveraging Sentence-BERT, the model is capable of capturing subtle nuances in sentiment, including sarcasm, irony, and ambiguity, that would typically be challenging for word-based models. This level of understanding is particularly important in social media, where users often express emotions in creative and indirect ways.

Sentence-BERT provides a highly effective method for vectorizing sentences by generating dense, context-aware sentence embeddings. This technique allows for deep semantic analysis, enabling the model to understand the meaning of words in their specific context within a sentence. Unlike traditional word-based models, Sentence-BERT focuses on entire sentences, making it especially suitable for complex tasks like sentiment analysis in social media contexts. By leveraging this approach, the model can accurately classify sentiment even in the presence of informal language, slang, and ambiguous expressions, making it a powerful tool for analyzing large volumes of text data.

XGBoost processes categorical features without requiring explicit conversion into numerical values, thereby reducing the risk of information loss and improving efficiency. It is particularly effective in tasks such as text classification, including sentiment analysis, spam detection, and topic classification. When integrated with other natural language processing techniques like Sentence-BERT, XGBoost can leverage high-quality vector representations of texts, enabling more precise classification. Sentence-BERT models transform text data into vectors that store semantic information, which is then fed into the XGBoost model for further classification (Singh, 2025).

As a result, XGBoost effectively classifies texts based on their semantic content. The model provides a range of hyperparameters—such as the number of trees, tree depth, learning rate, and others—that can be fine-tuned for optimal performance. Even when working with large datasets, XGBoost maintains high performance, making it particularly useful for analyzing extensive data in this context (Wang, 2025). The model's functionality is illustrated in Fig. 2.

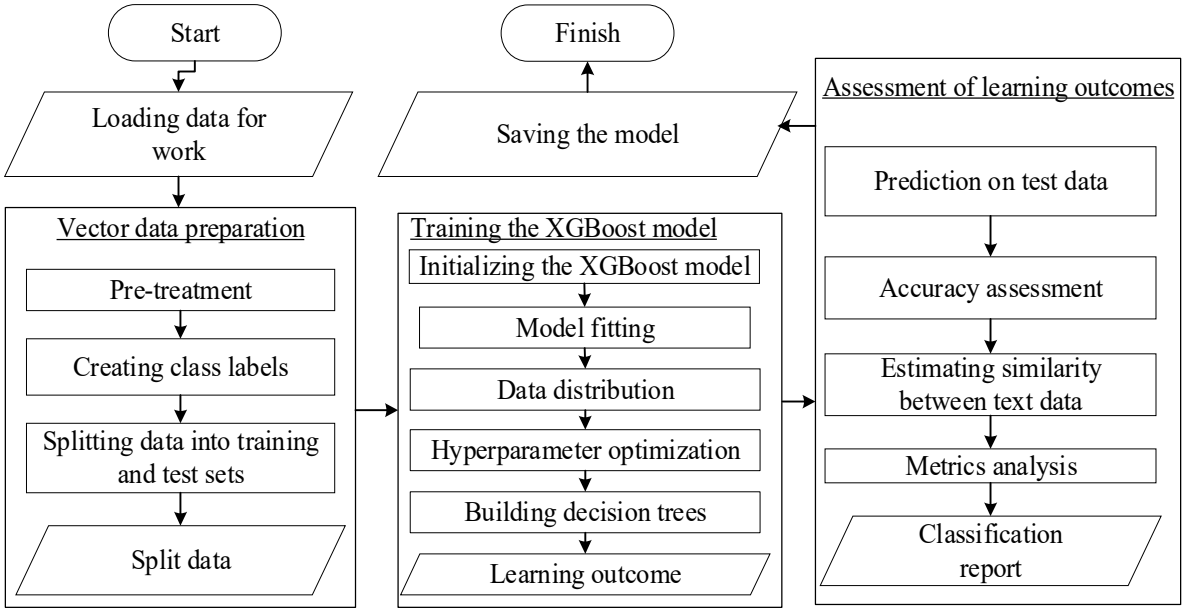

Fig. 2. Algorithm of the XGBoost model

After training the model, it is essential to evaluate its performance on the test set. To accomplish this, various metrics, such as accuracy, the confusion matrix, and the classification report, are employed. These metrics provide insight into how well the model classifies messages based on their mood and help determine whether the model can accurately predict positive or negative moods. For a more in-depth analysis of the results, the Sentence Embedding Similarity metric is used. This metric allows for the measurement of similarity between message vectors, which is particularly valuable in tasks that require not only classification but also the detection of similarities between texts that may share similar content but exhibit different moods. This approach facilitates a deeper analysis of the texts, enhancing the overall quality of classification. The block diagram illustrating the algorithm of the intelligent system is presented in Fig. 3.

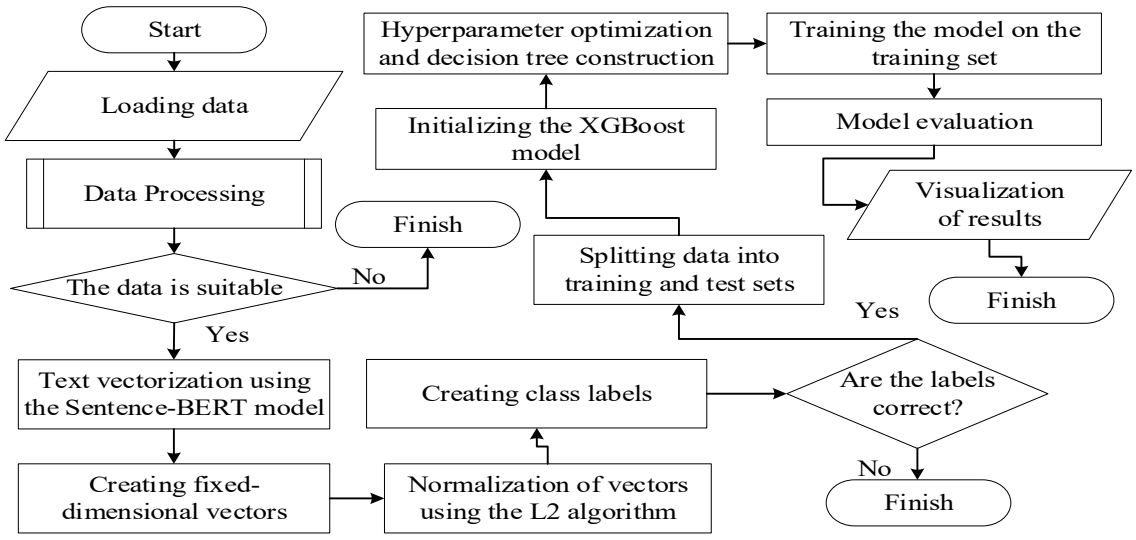

Fig. 3. Scheme of the information system algorithm

The study of the Sentiment140 dataset using Sentence-BERT, XGBoost, and Sentence Embedding Similarity methods enables the development of an effective information system for classifying messages based on sentiment. The key stages include preprocessing of text data, vectorization and normalization, training the model on numerical features, and evaluating its performance using various metrics. An important aspect of the process is also the analysis of similarity between messages, which enhances the quality of classification and ensures the high accuracy of the model.

Main Results

For the study of the Sentiment140 dataset, Python 3.12 and the PyCharm integrated development environment were selected. The initial step involves downloading the necessary dataset, which contains data from Twitter, with each user message labeled to indicate its emotional mood. The Sentiment140 dataset includes the following features: the text of the message (a text feature—message content on the social network, with a limit of 280 characters), the emotional mood of the message (the target feature), as well as additional parameters such as the time of publication and other metadata.

The goal of the study is to explore the relationships between text features and emotional mood in the context of social media. The primary objective is to develop a machine learning model capable of classifying messages by mood (positive or negative) based on their text content. Data analysis will allow an assessment of the model's effectiveness and its ability to accurately predict users' emotional moods based on text data, which is valuable for real-time sentiment analysis across large volumes of text. A fragment of the downloaded dataset is shown in Fig. 4.

1	0,"1467810369","Mon Apr 06 22:19:45 PDT 2009","NO_QUERY","_TheSpecialOne_","@switchfoot http://twitpic.com/2y1zl - Awww, that's a bummer. You shou
2	0,"1467810672","Mon Apr 06 22:19:49 PDT 2009","NO_QUERY","scotthamilton","is upset that he can't update his Facebook by texting it... and might cry as a res
3	0,"1467810917","Mon Apr 06 22:19:53 PDT 2009","NO_QUERY","mattycus","@Kenichan I dived many times for the ball. Managed to save 50% The rest go out o
4	0,"1467811184","Mon Apr 06 22:19:57 PDT 2009","NO_QUERY","ElleCTF","my whole body feels itchy and like its on fire "
5	0,"1467811193","Mon Apr 06 22:19:57 PDT 2009","NO_QUERY","Karoli","@nationwideclass no, it's not behaving at all. i'm mad. why am i here? because I can't s
6	0,"1467811372","Mon Apr 06 22:20:00 PDT 2009","NO_QUERY","joy_wolf","@Kwesidei not the whole crew "
7	0,"1467811592","Mon Apr 06 22:20:03 PDT 2009","NO_QUERY","mybirch","Need a hug "
8	0,"1467811594","Mon Apr 06 22:20:03 PDT 2009","NO_QUERY","coZZ","@LOLTrish hey long time no see! Yes.. Rains a bit ,only a bit LOL , I'm fine thanks , how'
9	0,"1467811795","Mon Apr 06 22:20:05 PDT 2009","NO_QUERY","2Hood4Hollywood","@Tatiana_K nope they didn't have it "
10	0,"1467812025","Mon Apr 06 22:20:09 PDT 2009","NO_QUERY","mimismo","@twittera que me muera ? "
11	0,"1467812416","Mon Apr 06 22:20:16 PDT 2009","NO_QUERY","erinx3leannexo","spring break in plain city... it's snowing "
12	0,"1467812579","Mon Apr 06 22:20:17 PDT 2009","NO_QUERY","pardonlauren","I just re-pierced my ears "
13	0,"1467812723","Mon Apr 06 22:20:19 PDT 2009","NO_QUERY","TLeC","@caregiving I couldn't bear to watch it. And I thought the UA loss was embarrassing . . .
14	0,"1467812771","Mon Apr 06 22:20:19 PDT 2009","NO_QUERY","robrobbierobert","@octolinz16 It it counts, idk why I did either. you never talk to me anymore
15	0,"1467812784","Mon Apr 06 22:20:20 PDT 2009","NO_QUERY","bayofwolves","@smarrison i would've been the first, but i didn't have a gun. not really thoug
16	0,"1467812799","Mon Apr 06 22:20:20 PDT 2009","NO_QUERY","HairByJess","@iamjazzyfizzle I wish I got to watch it with you!! I miss you and @iamlilnicki how
17	0,"1467812964","Mon Apr 06 22:20:22 PDT 2009","NO_QUERY","lovesongwriter","Hollis' death scene will hurt me severely to watch on film wry is directors cut
18	0,"1467813137","Mon Apr 06 22:20:25 PDT 2009","NO_QUERY","armotley","about to file taxes "
19	0,"1467813579","Mon Apr 06 22:20:31 PDT 2009","NO_QUERY","starkissed","@LettyA ahh ive always wanted to see rent love the soundtrack!!"
20	0,"1467813782","Mon Apr 06 22:20:34 PDT 2009","NO_QUERY","gi_gi_bee","@FakerPattyPattz Oh dear. Were you drinking out of the forgotten table drinks? "
21	0,"1467813985","Mon Apr 06 22:20:37 PDT 2009","NO_QUERY","quanvu","@alydesigns i was out most of the day so didn't get much done "

Fig. 4. Fragment of the downloaded dataset

Prior to training the machine learning model, performing thorough data preprocessing is crucial to ensure the quality and consistency of the dataset. This preprocessing process involves multiple steps, each designed to refine the data and prepare it for analysis and model training. The first step in preprocessing is text cleaning, where all irrelevant elements are removed from the text data. Special characters, URLs, mentions (such as “@username”), and hashtags (such as “#hashtag”) are discarded, as they do not contribute meaningful information for sentiment analysis.

This helps streamline the text, making it easier to focus on the actual content of the messages. Additionally, converting all text to lowercase is an essential part of this step. Lowercasing ensures that words like “Happy” and “happy” are treated as the same word, thus standardizing the text and reducing redundancy in the dataset. The next step is tokenization, which involves breaking the text down into smaller units, typically individual words or tokens. This step transforms the raw text into a structured format that can be easily processed by the machine learning model. Tokenization allows the model to understand the text at the word level, which is vital for extracting meaningful patterns and features related to sentiment.

Once tokenization is complete, the next stage is stop-word removal. Stop-words are common, high-frequency words such as “and”, “the”, “is”, and “in”, which do not carry significant meaning in the context of sentiment analysis. These words are filtered out because they tend to occur in almost every sentence and do not provide any relevant information about the sentiment or emotion expressed in the text. Removing stop-words helps reduce noise in the data and improves the efficiency of the model. Following stop-word removal, lemmatization is performed. Lemmatization is the process of reducing words to their base or root form. For example, the word "running" would be converted to its base form “run”, and “better” would become “good”. This step ensures uniformity in the dataset, as different word forms (e.g., “runs”, “running”, “runner”) are treated as the same word. Lemmatization is preferred over stemming because it produces more meaningful and linguistically correct results, preserving the semantic meaning of words.

After completing these preprocessing steps, the dataset is ready for analysis. In the case of sentiment analysis, the Sentiment140 dataset is typically divided into two primary sentiment classes: negative and positive. Each message in the dataset is labeled with one of these two sentiments based on the emotional tone conveyed by the text. The negative class contains messages that express negative emotions, such as sadness, anger, or frustration, while the positive class contains messages that express positive emotions, such as happiness, excitement, or satisfaction. This binary classification approach enables the model to focus on detecting and categorizing emotional sentiment from the text data. At this point, the data is ready for further analysis and model training.

The accuracy assessment of the model after training yielded high results, demonstrating its effectiveness in classifying text data. The model achieved an overall accuracy of 90 % on the test set. Analyzing the individual metrics for each class revealed balanced performance. For the negative class, the model showed an accuracy of 91 %, a recall of 87 %, and an F1-score of 89 %. These results suggest that the model effectively identifies negative examples, although some misclassification occurred. For the positive class, the accuracy was 89 %, recall was 92%, and the F1-score was 91 %.

The model exhibited a slightly better ability to detect positive classes while maintaining a strong balance between accuracy and recall. The summary metrics revealed an average macro score (macro avg) for accuracy, recall, and F1-score of 90 %, indicating uniform performance across classes. The weighted average score (weighted avg), which accounts for the frequency of each class in the data, also stood at 90 %, underscoring the model's stable performance regardless of class distribution. The training accuracy, recall, and F1-scores are shown in Fig. 5.

Training set accuracy: 0.92

	precision	recall	f1-score	support
negative	0.92	0.93	0.93	240525
positive	0.93	0.92	0.92	239475
accuracy			0.92	480000
macro avg	0.92	0.92	0.92	480000
weighted avg	0.92	0.92	0.92	480000

Test set accuracy: 0.90

	precision	recall	f1-score	support
negative	0.90	0.90	0.90	240448
positive	0.90	0.90	0.90	239552
accuracy			0.90	480000
macro avg	0.90	0.90	0.90	480000
weighted avg	0.90	0.90	0.90	480000

Fig. 5. Information about the created neural network model

The classification model results exhibited high accuracy and balance in predicting negative and positive classes. The AUC value for the ROC curve was 0.88, signifying a high degree of class separation. The ROC curves demonstrated stable performance of the model across varying classification thresholds. Precision-Recall analysis for the negative class showed an increase in precision with a decrease in recall, reaching a maximum of 1.0 at higher thresholds. Recall decreased with lower thresholds, maintaining stability at initial levels and gradually declining for less relevant samples.

For the positive class, the maximum accuracy of 1.0 was also achieved, indicating excellent detection under certain conditions. Completeness exhibited a steady decline but remained significant at medium thresholds. The numerical values of the confusion matrix and the Precision-Recall analysis results are shown in Fig. 6, and the confusion matrix is depicted in Fig. 7.

```
Confusion matrix for the training set:Confusion matrix for the test set:
          negative  positive                      negative  positive
negative    222596     17929          negative    216191     24257
positive     18113    221362          positive     23864    215688
Precision: [0.50164846 0.50164691 0.50164847 ... 1.       1.       1.       ]
Recall: [1.00000000e+00 9.99993770e-01 9.99993770e-01 ... 1.24591185e-05
 6.22955926e-06 0.00000000e+00]

Precision for class negative: [0.50164062 0.50163907 0.50163751 ... 0.        0.        1.
Recall for class negative: [1.        0.99999377 0.99998754 ... 0.        0.        0.

Precision for class positive: [0.50164846 0.50164691 0.50164847 ... 1.        1.        1.
Recall for class positive: [1.00000000e+00 9.99993770e-01 9.99993770e-01 ... 1.24591185e-05
 6.22955926e-06 0.00000000e+00]
```

Fig. 6. Change in precision and recall during training model and final confusion matrix

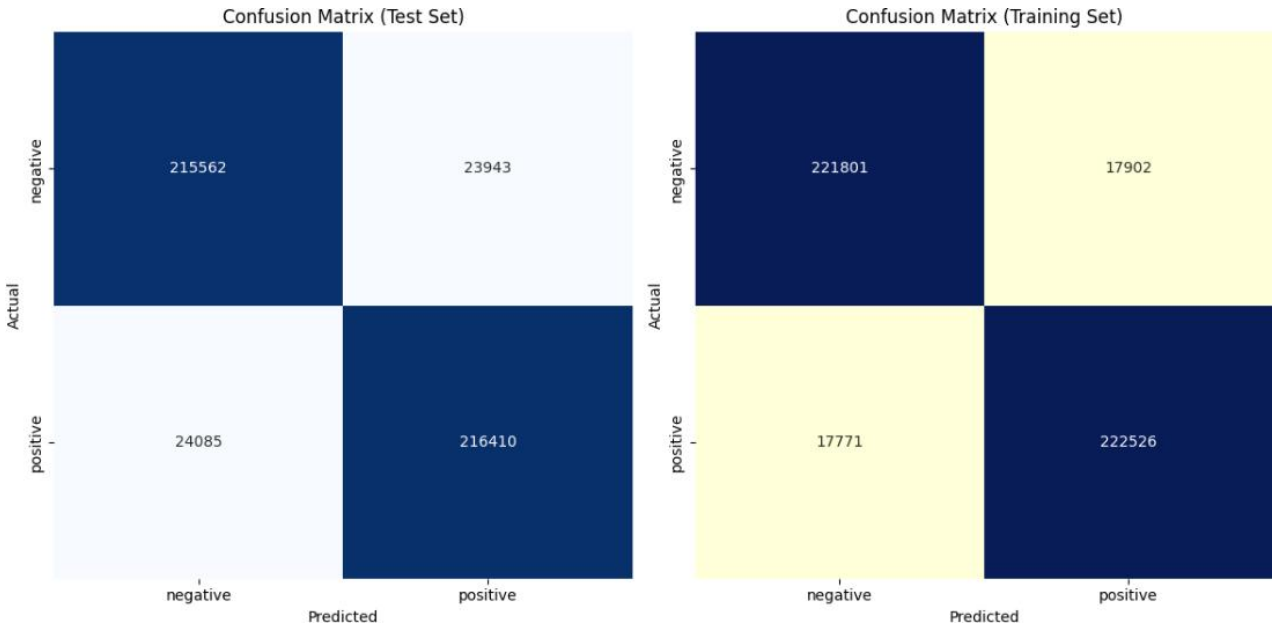

Fig. 7. Ratio of correct and incorrect predictions on the confusion matrix

These results demonstrate that the model can accurately identify both classes in text data, offering high performance even on balanced datasets. The model's effectiveness highlights its potential for application in text classification tasks and sentiment analysis on large text datasets. The ROC curve in Fig. 8 illustrates the model’s ability to distinguish between tweets with negative and positive sentiment. The high AUC value (0.90) indicates the model’s proficiency in recognizing sentiments with high accuracy.

A larger area under the curve signifies that the model minimizes the total number of false positive and false negative errors. A larger AUC value signifies that the model minimizes the total number of false positives and false negatives. A false positive occurs when the model incorrectly labels a negative tweet as positive. Minimizing false positives ensures that negative sentiment isn't mistakenly classified as positive, which is important for tasks like sentiment analysis, where accurate sentiment identification is crucial. A false negative occurs when the model incorrectly labels a positive tweet as negative. Minimizing false negatives ensures that positive sentiment isn't overlooked or misrepresented, which possibly could impact decision-making based on sentiment analysis.

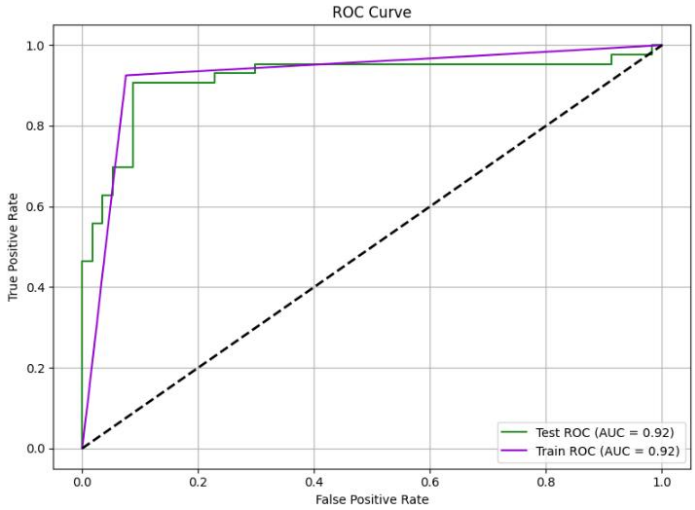

Fig. 8. ROC curve for separating positive and negative classes of user messages

The Precision-Recall curve in Fig. 9 shows the trade-off between precision and recall for classifying positive and negative tweets, depending on the threshold value. High precision means that most tweets classified by the model as positive or negative actually belong to the corresponding class, which is crucial for minimizing false positives. High recall indicates that the model can identify all tweets of the corresponding sentiment in the test sample, thereby reducing false negative predictions. The Sentiment140 dataset consists of short texts (user messages on Twitter), which often contain emotionally charged words, abbreviations, emojis, or other nonlinear language constructs, making classification more challenging. In this context, the Precision-Recall curve demonstrates that the model performs effectively, providing reliable predictions.

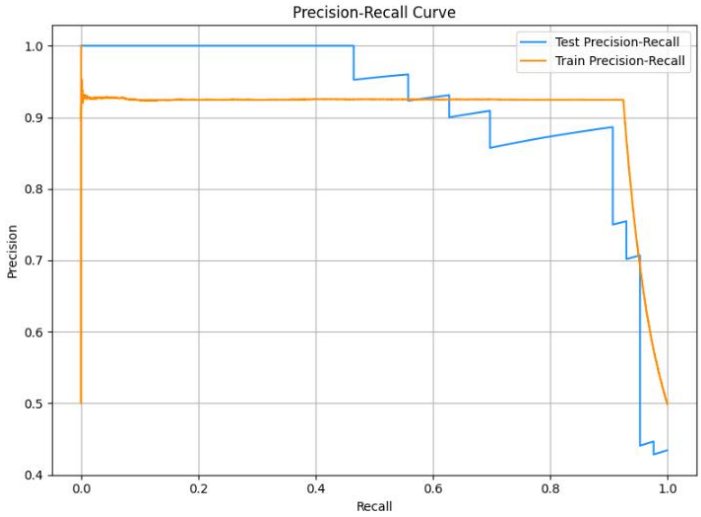

Fig. 9. The ratio of accuracy to completeness

The calibration curve shown in Fig. 10 is an important tool for assessing the alignment between the predicted probabilities of the model and the actual proportion of positive results in the test sample.

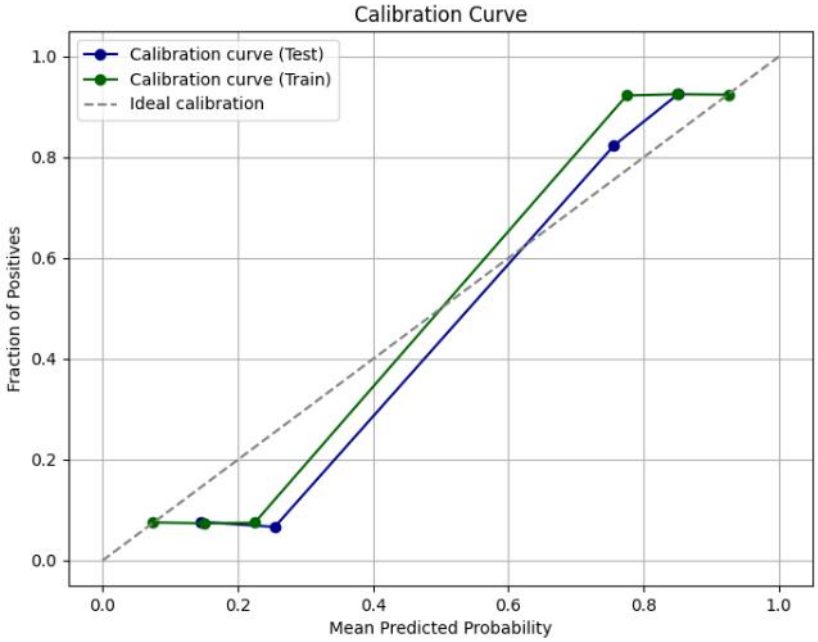

Fig. 10. Calibration curve for consistency assessment

The model demonstrates high accuracy for predicted probabilities in the range [0.7579–0.8490], where the actual positive fraction closely correlates with the predictions of 0.9286 and 0.8519, respectively. This indicates the robustness of the model for confident predictions. In the lower intervals ([0.1449–0.2499]), an underestimation of the positive fraction is observed, suggesting the need for further calibration of the model to improve its performance in these ranges.

Cosine similarity is a key tool for quantifying semantic similarity between text data. In this study, the cosine similarity matrix between five tweets was computed using Sentence-BERT vectorization. The cosine similarity values range from 0.636 to 1.000, as shown in Figs. 11 and 12. Higher values indicate a high degree of semantic similarity between the corresponding texts, which may suggest similarity in the topic or key expressions.

The results of the cosine similarity matrix confirm the effectiveness of Sentence-BERT in detecting semantic similarity between texts. High values for certain pairs of tweets (e.g., Tweet 3 and Tweet 4) show that the model correctly identifies content proximity, even if the texts differ at the level of syntax or surface structure. Lower values (e.g., Tweet 2 and Tweet 5) may indicate texts that are thematically less related or differ significantly in expression, which is crucial for analyzing data diversity.

```
Cosine Similarity Matrix between the first 5 tweets:
          Tweet 1   Tweet 2   Tweet 3   Tweet 4   Tweet 5
Tweet 1  1.000000  0.694885  0.773801  0.748353  0.732547
Tweet 2  0.694885  1.000000  0.710615  0.793654  0.738366
Tweet 3  0.773801  0.710615  1.000000  0.793794  0.800139
Tweet 4  0.748353  0.793654  0.793794  1.000000  0.774994
Tweet 5  0.732547  0.738366  0.800139  0.774994  1.000000
Calibration Curve Data:
Mean Predicted Probabilities: [0.14988771 0.24993101 0.74999233 0.84994613]
Fraction of Positives: [0.10219866 0.10189194 0.90100211 0.89927636]
```

Fig. 11. Cosine similarity matrix between the first 5 tweets

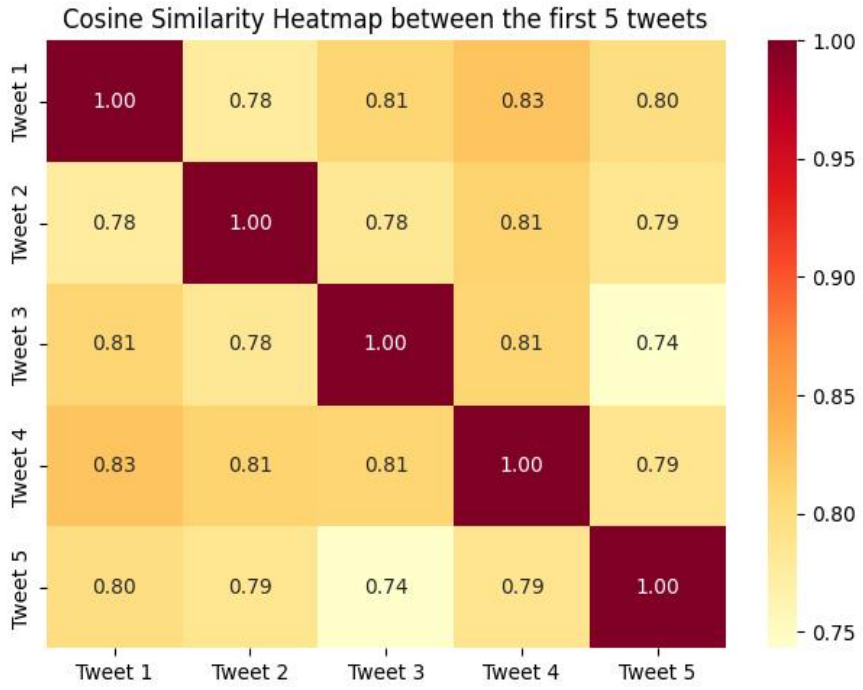

Fig. 12. Heatmap of cosine similarity between the first 5 tweets

The F1-measure is an important metric in classification problems that combines both accuracy and completeness into a single value, providing a comprehensive understanding of the model's performance, as depicted in Fig. 13. In this study, the F1-measure was evaluated for different threshold values in the range from 0 to 1, with a step size of 0.0204. The highest F1-measure, 0.9074, was achieved for thresholds ranging from 0.306 to 0.673, indicating effective model performance within this range. This confirms the model's ability to accurately identify both positive and negative classes under conditions of a balanced ratio of accuracy and completeness. Maintaining a high F1-measure within the middle range of thresholds (0.306–0.673) further highlights the model's stability to changes in threshold values, which is an important aspect of its reliability.

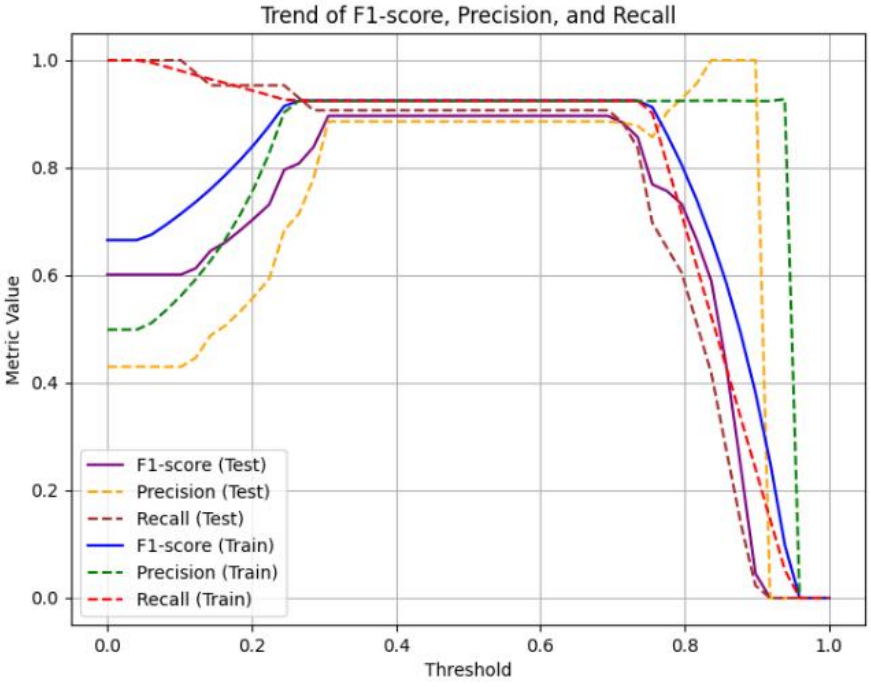

Fig. 13. Trends in F1-measure, accuracy and completeness

Cumulative growth is a key characteristic that allows the progressive change in model performance to be assessed when analyzing classification problems. In this study, Figure 14 illustrates the sequential accumulation of correctly classified examples to determine the dynamics of model performance. Stable linear growth in the middle range emphasizes the consistency and reliability of the model, while the formation of a plateau in the upper range signals the point at which the model's capabilities have been reached, which is typical for tasks of high complexity.

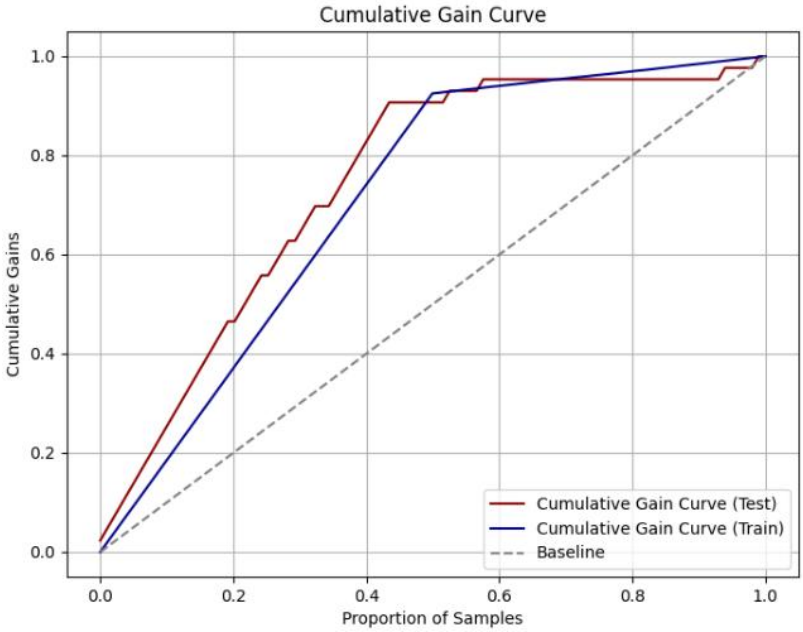

Fig. 14. Cumulative growth curve

The results provide a detailed evaluation of the model's performance using several key metrics that offer insights into its classification abilities, strengths, and potential areas for improvement. The confusion matrix plays a crucial role in revealing how well the model performs across different sentiment classes. It provides a clear breakdown of the model's predictions into true positives, true negatives, false positives, and false negatives. These values are essential for assessing the model's overall accuracy and understanding where it might be making errors. By analyzing these values, one can pinpoint specific areas where the model might need adjustments, such as ability to identify positive sentiments.

The ROC curve offers additional insights into the model's performance, particularly in terms of its ability to balance sensitivity (the true positive rate) and specificity (the true negative rate). A well-balanced ROC curve suggests that the model can effectively distinguish between positive and negative instances without a bias toward either class. The ROC curve is particularly useful when analyzing a model's performance across different classification thresholds, offering a visual representation of how changes in threshold impact the trade-off between sensitivity and specificity. A key metric here is the, which quantifies the model's classification ability. A higher AUC value indicates better overall performance, as it suggests that the model is more adept at distinguishing between positive and negative instances.

The Precision-Recall curve becomes especially valuable when dealing with imbalanced datasets, where one class might be underrepresented. Unlike the ROC curve, which can sometimes present an overly optimistic view in such cases, the Precision-Recall curve focuses on the model's ability to correctly identify the positive class. This is particularly important in sentiment analysis, where positive sentiment might be less frequent. The Precision-Recall curve provides a more detailed view of how well the model handles the minority class, ensuring that the model doesn't simply predict the majority class in most cases. High precision indicates that the model's positive predictions are accurate, while high recall shows that it successfully identifies most of the positive instances. Together, these metrics ensure that the model does not neglect the less frequent positive class.

The calibration curve offers another valuable perspective, providing a visual representation of how well the model's predicted probabilities align with the true probabilities. In many applications, particularly those involving decision-making based on predicted probabilities, this alignment is crucial. A well-calibrated model will produce probability estimates that closely match the true likelihood of an event occurring. The cosine similarity matrix is another important tool in this evaluation, providing a measure of the semantic similarity between different text samples. It allows the model to understand the degree of similarity between documents by comparing their sentence embeddings. This matrix is useful for a variety of tasks, including text classification, clustering, and recommendation systems. It highlights how well the model captures the underlying meaning of the text rather than relying solely on surface-level features like word frequency.

Finally, the analysis of the F1-measure, accuracy, and completeness across different stages of training and testing provides a comprehensive view of the model's stability and consistency. The F1-measure is particularly useful for balancing precision and recall, providing a single metric that reflects the model's ability to accurately classify both positive and negative instances. Together, these evaluation metrics suggest that the model is capable of accurately classifying emotional content in text, while offering the reliability and versatility needed for real-world deployment. These results reinforce the model's potential for applications in sentiment analysis, opinion mining, and content categorization, ensuring it can perform consistently and accurately across different types of text data.

Conclusions

This study makes a significant contribution to the field of sentiment analysis and emotional content classification by introducing a novel approach that combines Sentence-BERT, XGBoost, and Sentence Embedding Similarity. While previous researches has explored the use of Sentence-BERT and XGBoost for sentiment analysis, the novelty of this work lies in the innovative integration of these methods, particularly with a focus on large-scale sentiment analysis tasks using the Sentiment140 dataset.

This research uniquely explores the influence of specific vocabulary on classification outcomes, while also leveraging Sentence Embedding Similarity to enhance the model's ability to capture semantic relationships between text samples. The core of the novelty lies in the Sentence-BERT model, which generates context-aware sentence embeddings, providing a richer understanding of the text by capturing contextual relationships and semantic nuances. By transforming text into dense vector representations that encapsulate the meaning of entire sentences, Sentence-BERT overcomes the limitations of traditional word embeddings, making it ideal for analyzing the often informal, fragmented, and slang-laden language typical of social media content. This capability is critical for emotional content classification, where capturing the full sentiment of a sentence is essential.

The integration of XGBoost, a powerful gradient boosting algorithm, enhances the model's performance by effectively handling complex, high-dimensional data and providing high classification accuracy. Combined with Sentence Embedding Similarity, which measures the semantic similarity between text samples, the model gains the ability to detect subtle emotional connections between text samples, even when those connections are not immediately apparent. This opens up new possibilities for clustering and detecting duplicate content in large text datasets, as well as improving sentiment classification across various types of textual expressions.

The performance of the model was thoroughly evaluated using key metrics such as the ROC curve and Precision-Recall curves, which confirmed its ability to classify text across a wide range of sentiment categories with high reliability. Notably, the model demonstrated a strong balance between precision and recall, minimizing the omission of important positive examples while maintaining accuracy. The calibration curves further validated the model's ability to make trustworthy probabilistic predictions, which is crucial for real-world applications. An important extension of the analysis involved cosine similarity, which highlighted the model's capability to detect semantic similarity between seemingly unrelated text samples. This feature is invaluable for tasks such as clustering and identifying duplicate content, as it allows the model to group text samples based on underlying emotional or thematic similarities. By adjusting the classification threshold, the model achieved an optimal balance between detecting true positives and minimizing false positives, ensuring its practical applicability in real-world scenarios.

In conclusion, this study demonstrates the effectiveness of combining Sentence-BERT, XGBoost, and Sentence Embedding Similarity for classifying emotional content in text data. This novel methodology offers a robust and scalable solution for large-scale sentiment analysis tasks, with promising applications in user feedback analysis, public opinion research, automated content moderation, and trend detection on social networks. Future research could explore adapting this approach to other datasets, integrating additional sources of information, and refining the model to address challenges like imbalanced data. These directions offer exciting opportunities for further enhancing the model's effectiveness and broadening its use across various interdisciplinary research domains.

REFERENCES

1. Aggarwal, P., & Mahajan, R. (2024). Shielding Social Media: BERT and SVM Unite for Cyberbullying Detection and Classification. *Journal of Information Systems and Informatics*, 6(2), 607–623. DOI: https://doi.org/10.51519/journalisi.v6i2.692
2. Al-Qudah, D. A., Al-Zoubi, A. M., Cristea, A. I., Merelo-Guervós, J. J., Castillo, P. A., & Faris, H. (2025). Prediction of sentiment polarity in restaurant reviews using an ordinal regression approach based on evolutionary XGBoost. *PeerJ Computer Science*, 11, e2370–e2370. DOI: https://doi.org/10.7717/peerj-cs.2370
3. Atmaja, A. I., Maimunah, M., & Sukmasetya, P. (2024). Analysis of Labeling and Class-Balancing Effects on Clash of Champions Sentiment Using LSTM and BERT. *Journal of Information Systems and Informatics*, 6(4), 2868–2891. DOI: https://doi.org/10.51519/journalisi.v6i4.929
4. Aziz, K., Ji, D., Chakrabarti, P., Chakrabarti, T., Iqbal, M. S., & Abbasi, R. (2024). Unifying aspect-based sentiment analysis BERT and multi-layered graph convolutional networks for comprehensive sentiment dissection. *Scientific Reports*, 14(1). DOI: https://doi.org/10.1038/s41598-024-61886-7
5. Batiuk, T., & Dosyn, D. (2023). Intellectual system for clustering users of social networks derived from the message sentiment analysis. *Journal of Lviv Polytechnic National University* "*Information Systems and Networks*", 13, 121–138. DOI: https://doi.org/10.23939/sisn2023.13.121
6. Batiuk, T., & Dosyn, D. (2024). Realization of the decision-making support system for twitter users' publications analysis. *Radio Electronics Computer Science Control*, 1(24), 175-187. DOI: https://doi.org/10.15588/1607-3274-2024-1-16
7. He, L. (2024). Enhanced twitter sentiment analysis with dual joint classifier integrating RoBERTa and BERT architectures. *Frontiers in Physics*, 12. DOI: https://doi.org/10.3389/fphy.2024.1477714
8. Ivokhin, E., & Oletsky, O. (2022). Restructuring of the Model "State–Probability of Choice" Based on Products of Stochastic Rectangular Matrices. *Cybernetics and Systems Analysis*, 58(2), 242-250. DOI: https://doi.org/10.1007/s10559-022-00456-z
9. Khan, A., Majumdar, D., & Mondal, B. (2025). Sentiment analysis of emoji fused reviews using machine learning and Bert. *Scientific Reports*, 15(1). DOI: https://doi.org/10.1038/s41598-025-92286-0
10. Najeem Olawale Adelakun, & Abimbola Baale Adebisi. (2024). Sentiment analysis of financial news using the BERT model. *ITEGAM-Journal of Engineering and Technology for Industrial Applications (ITEGAM-JETIA)*, 10(48). DOI: https://doi.org/10.5935/jetia.v10i48.1029
11. Ogunleye, B., Sharma, H., & Shobayo, O. (2024). Sentiment Informed Sentence BERT-Ensemble Algorithm for Depression Detection. *Big Data and Cognitive Computing*, 8(9), 112. DOI: https://doi.org/10.3390/bdcc8090112
12. Oletsky, O. (2021). Exploring Dynamic Equilibrium Of Alternatives On The Base Of Rectangular Stochastic Matrices. *Modern Machine Learning Technologies and Data Science Workshop*, MoMLeT&DS 2021, 5-6 June 2021, Lviv-Shatsk, Ukraine, 2917, 151-160. http://ceur-ws.org/Vol-2917/
13. Roumeliotis, K. I., Tselikas, N. D., & Nasiopoulos, D. K. (2024). Leveraging Large Language Models in Tourism: A Comparative Study of the Latest GPT Omni Models and BERT NLP for Customer Review Classification and Sentiment Analysis. *Information*, 15(12), 792. DOI: https://doi.org/10.3390/info15120792
14. Setiawan, M. J., & Vinna Rahmayanti Setyaning Nastiti. (2024). DANA App Sentiment Analysis: Comparison of XGBoost, SVM, and Extra Trees. *Jurnal Sisfokom (Sistem Informasi Dan Komputer)*, 13(3), 337–345. DOI: https://doi.org/10.32736/sisfokom.v13i3.2239
15. Singh, D., Barve, S., & Dwivedi, A. K. (2025). OptiASAR: Optimized Aspect Sentiment Analysis with BiLSTM-GRU and NER-BERT in Healthcare Decision-making. *IEEE Access*, 1–1. DOI: https://doi.org/10.1109/access.2025.3549303
16. Wang, Z. (2025). Sentiment Analysis of Mobile Phone Reviews Using XGBoost and Word Vectors. *ITM Web of Conferences*, 70, 03018. DOI: https://doi.org/10.1051/itmconf/20257003018

ІНТЕЛЕКТУАЛЬНИЙ АНАЛІЗ ТЕКСТОВИХ ДАНИХ У СОЦІАЛЬНИХ МЕРЕЖАХ ІЗ ВИКОРИСТАННЯМ BERT І XGBOOST

Тарас Батюк[1], Дмитро Досин[2]

[1, 2] Національний університет „Львівська політехніка”,
кафедра інформаційних систем та мереж, Львів, Україна
[1] E-mail: taras.m.batiuk@lpnu.ua, ORCID: 0000-0001-5797-594X
[2] E-mail: dmytro.h.dosyn@lpnu.ua, ORCID: 0000-0003-4040-4467

У цій статті представлено комплексний підхід до аналізу настроїв у соціальних мережах із застосуванням сучасних методів опрацювання тексту та алгоритмів машинного навчання. Основний фокус — інтеграція моделі Sentence-BERT для векторизації тексту та XGBoost для класифікації настроїв. Використовуючи набір даних Sentiment140, було проведено широке дослідження текстових повідомлень, позначених анотаціями настроїв. Модель Sentence-BERT дозволяє генерувати високоякісні векторні представлення текстових даних, зберігаючи як лексичні, так і контекстуальні зв’язки між словами. Це сприяє більш точному семантичному розумінню повідомлень, тим самим підвищуючи ефективність класифікації. Результати дослідження демонструють високу ефективність запропонованої моделі, досягнення загальної точності класифікації 90 %. Площа під кривою ROC (AUC) 0,88 додатково підтверджує здатність моделі ефективно розрізняти класи настрою. Аналіз кривої Precision-Recall підкреслює міцний баланс між точністю та запам’ятовуванням, що особливо важливо для опрацювання незбалансованих наборів даних. Крім того, калібрувальні криві вказують на високий ступінь узгодженості між прогнозованими ймовірностями та фактичними результатами, тоді як матриця косинусної подібності підтверджує здатність моделі фіксувати семантичну близькість між текстами. Окрім класифікації, у дослідженні також розглядається показник F1 на різних порогових рівнях, що дозволяє визначити оптимальний робочий діапазон для моделі. Діаграма сукупного посилення ілюструє поступове покращення продуктивності класифікації, підкреслюючи стабільність моделі під час опрацювання великомасштабних текстових даних. Запропонований підхід служить універсальним інструментом для аналізу настроїв, кластеризації тексту та ідентифікації трендів у соціальних мережах. Результати цього дослідження мають практичне значення в таких сферах, як маркетинг, аналіз громадської думки, автоматизована модерація вмісту та прогнозування соціальних тенденцій.

Ключові слова – Sentence-BERT, XGBoost, векторизація тексту, трансформери, косинусна схожість.