Evaluation of the effectiveness of machine learning methods for detecting disinformation in Ukrainian text data

Khrystyna Lipianina-Honcharenko[1], Mariana Soia[1], Khrystyna Yurkiv[1], Andrii Ivasechko[1].
[1] *West Ukrainian National University, Lvivska str., 11, Ternopil, 46000, Ukraine*

Abstract
In today's world, where information spreads with unprecedented speed, disinformation poses a serious challenge to public trust and information security. The full-scale invasion of Ukraine by Russia in 2022 activated the use of disinformation as a tool of hybrid warfare, highlighting the need for effective methods of identification and control. This article focuses on evaluating the effectiveness of various machine learning methods for detecting disinformation in Ukrainian text data, using a dataset that includes news headlines collected during the conflict. The study encompasses the analysis of logistic regression, support vector machines (SVM), random forest, gradient boosting, KNN, decision trees, XGBoost, and AdaBoost. Model evaluation was performed using standard metrics: precision, recall, F1-score, overall accuracy, and confusion matrix. The results indicate significant potential for using machine learning in the fight against disinformation, particularly the random forest model demonstrated the highest effectiveness. The study emphasizes the importance of adapting and optimizing classifiers for the specific task of disinformation analysis, paving the way for further research in this field.

Keywords
Disinformation, machine learning, classification, text data.

1. Introduction

In the modern information space, a vast amount of data is generated daily, a significant portion of which is news content. In the context of increasing globalization and accessibility of information technologies, information spreads rapidly through the network, making it a powerful tool for influencing public opinion. However, this also paves the way for the mass dissemination of disinformation, which can have significant consequences for society, politics, and international relations. Navigating this flow of information and distinguishing reliable data from false has become an increasingly important task.

The war that began with Russia's full-scale invasion of Ukraine in February 2022 is a striking example of the use of disinformation as a weapon in hybrid warfare. This has created a need for the development of effective tools for analyzing and classifying informational content, with the goal of identifying and counteracting disinformation.

In this context, machine learning and natural language processing methods play a key role in the detection and analysis of fake news. The application of these technologies allows for the automation of the disinformation detection process, providing fast and efficient processing of large volumes of data. At the same time, the development of effective machine learning models for information classification requires a deep understanding of data specifics, preprocessing methods, and model optimization.

This article is devoted to the analysis of a dataset containing news headlines collected during the Russo-Ukrainian conflict, aimed at identifying disinformation. It considers the application of various machine learning methods, including logistic regression, support vector machines (SVM), random forest, gradient boosting, KNN, decision trees, XGBoost, and AdaBoost for the classification of text data. The concluding section is dedicated to evaluating the effectiveness of these models using standard evaluation metrics, including precision, recall, F1-score, overall

Proceedings Acronym: Proceedings Name, Month XX–XX, YYYY, City, Country
xrustya.com@gmail.com (K. Lipianina-Honcharenko); mariankasoa@gmail.com (M. Soia); kh.yurkiv@wunu.edu.ua (K. Yurkiv); Andrewivasechko@gmail.com (A. Ivasechko).
0000-0002-2441-6292 (K. Lipianina-Honcharenko); 0009-0001-3719-6460 (M.Soia); 0009-0007-4917-3251 (K. Yurkiv); 0009-0002-8623-7828 (A. Ivasechko).

accuracy, and the confusion matrix, which allows determining the most effective methods for combating disinformation in the context of information warfare.

2. Related Work

In contemporary research in the field of information security and media content analysis, the importance of detecting and analyzing disinformation, especially anti-vaccine content on social media platforms such as Twitter, has gained particular relevance. In [1], language-neutral models are developed for detecting such content on a large scale, using multifaceted representations of messages in networks. Meanwhile, [2] focuses on the challenges associated with pre-training graph neural networks for context-oriented detection of fake news, pointing out strategic and resource constraints. In [3], a comparative analysis of supervised and unsupervised machine learning algorithms for detecting fake news is conducted, demonstrating their performance, efficiency, and robustness.

Research covering the analysis of disinformation and public opinion on the Russo-Ukrainian War employs a variety of methodologies, including sentiment analysis, creation and analysis of datasets, and studies of the impact of war on language choice. One study models and clusters sentiment trends of different countries regarding the war [4], while another offers a detailed dataset of tweets related to the crisis [5]. The discourse on Twitter about the war is also analyzed, with a particular focus on language and the geographical origin of tweets [6]. Another study focuses on the challenges of labeling sensitive content and its psychological impact on annotators [7]. It is also examined how the war affects the language choice of Ukrainians on Twitter, analyzing changes in language preferences over time [8]. A separate study provides insights into the activity on subreddits related to the conflict, analyzing post volumes, comments, and the level of engagement [9]. Research [10] demonstrates that the application of the BERT model for fake news detection achieves an accuracy of 79.88% and an area under the ROC curve of 0.87, highlighting its potential in combating disinformation on social networks.

As confirmed by the aforementioned analysis, research in the field of disinformation detection, particularly fake news, is a significant direction in the context of information security and media content analysis. The need for the development and application of advanced technological solutions for effective fake news detection is particularly compelling. However, there is a noticeable lack of research focused on the analysis of disinformation in the Ukrainian language, which poses a challenge to the scientific community to expand the linguistic spectrum of research in this field. Considering this, the aim of this study is to develop intelligent methods for detecting disinformation, specifically fake news, with an emphasis on Ukrainian-language content. This will enhance the level of information security and ensure the integrity of the news space in the conditions of the modern information society.

3. Research methodology

3.1 Dataset Description

For data collection and processing, the dataset (Ukrainian language) [11] was used, which contains approximately 10,700 news headlines about the Russo-Ukrainian War, collected from February 24 to December 11, 2022, covering the period from the beginning of the full-scale invasion.

The dataset for the analysis of disinformation in the Ukrainian media space consists of two main files: "data_set_4.csv" (Table 1) and "news_data.csv" (Table 2), each containing news headlines classified as true ("True") or false ("False"). The "data_set_4.csv" file records 8,237 true and 2,498 false news items, while "news_data.csv" contains a significantly larger number of true news items — 48,006, compared to 2,024 false, reflecting a wide range of informational content collected for the study of the dissemination of disinformation during the Russo-Ukrainian War.

Both datasets (see Table 1,2) are used for the analysis of disinformation in the Ukrainian media space and include data that were collected from official and unofficial sources with the purpose of studying the spread of fake news in the context of the Russo-Ukrainian War.

Table 1
Structure of data_set_4.csv

Field	Description	Data Type
Text	News Headline	Text
Label	Label that marks the news as true ("True") or false ("False")	Boolean
Link	Link to the news source (available only in this file)	Text

Table 2
Structure of news_data.csv

Field	Description	Data Type
Text	News Headline	Text
Label	Classification of the news as true ("True") or false ("False")	Boolean

The graphs (Fig.1) of label distribution show that in both datasets, the number of true messages predominates over the false ones. For example, in the first dataset, the ratio of true messages to false is high, which indicates a focus on reliable information.

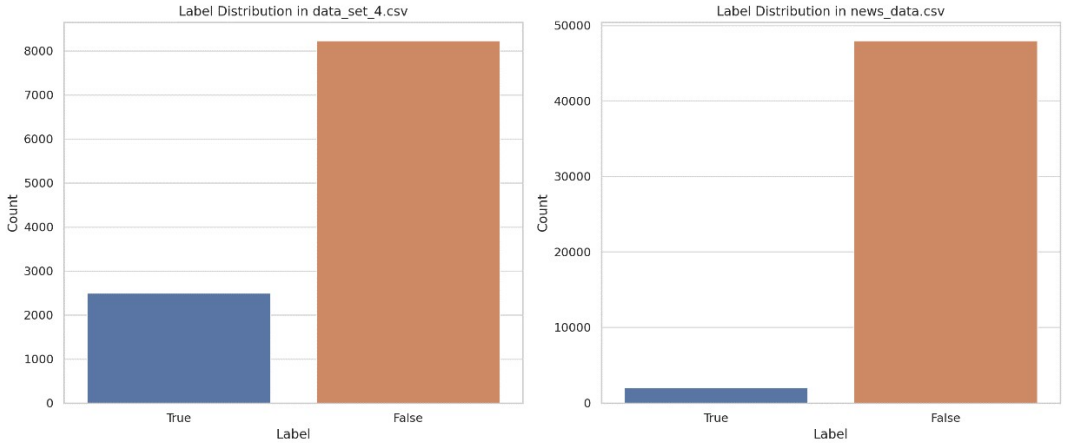

Figure 1: Label Distribution

The analysis of the most frequently used words (Fig.2) in the first dataset revealed words that appear hundreds of times. This allows identifying key topics of discussions or news, for example, the word "Ukraine" may occur most frequently, emphasizing the geographical or political focus of the collected data.

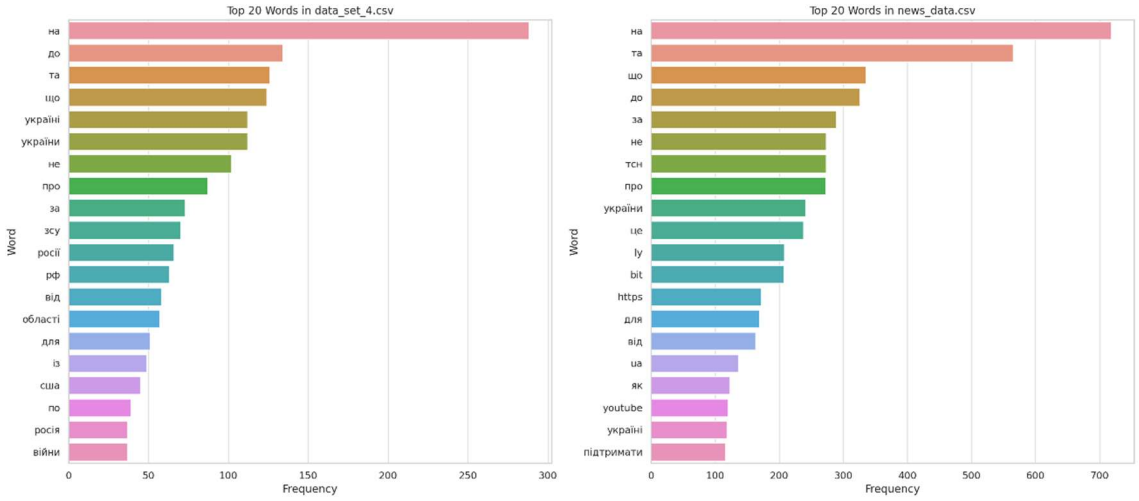

Figure 2: Top 20 Words

Most texts (Fig.3) in the first dataset have a length of 100 to 500 characters. Such information helps to understand the average volume of messages, which may indicate a prevalence of short news or overviews.

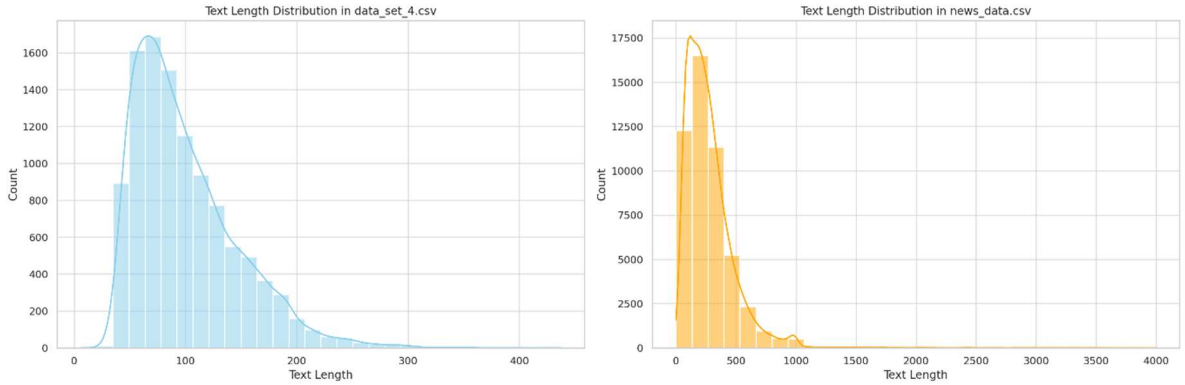

Figure 3: Distribution of Text Lengths

Analysis of the presented datasets covering headlines of news about the Russian-Ukrainian war demonstrates a significant advantage of credible information compared to false, emphasizing the focus on quality content. Considering that the dataset "news_data.csv" contains substantially more records compared to "data_set_4.csv", it is logical to use the former for training machine learning models as it provides a broader range of information and a greater number of examples for training. Meanwhile, the smaller dataset "data_set_4.csv" can serve as an excellent set for testing and evaluating the effectiveness of models on a smaller but specific data sample. This will allow assessing the model's ability to generalize learning on new, previously unknown data, emphasizing its practical value in real-world disinformation analysis conditions.

3.2 Description of used classifiers

In modern data analysis, especially in the context of detecting misinformation, the use of machine learning algorithms for classifying textual data becomes a key tool for developers and analysts. The diversity of classification methods [12], such as logistic regression, support vector machines (SVM), random forest, gradient boosting, k-nearest neighbors (KNN), decision tree, XGBoost, and AdaBoost, provides a wide range of approaches for data analysis and classification. Each of these algorithms has its unique advantages and limitations, making them more or less suitable for specific types of data and analytical tasks. The main goal is to select the optimal classifier that best fits the specificity of the task and the data we are working with.

Logistic regression [13] is a classification method used to predict the probability of two possible outcomes based on one or more independent variables. It transforms the linear combination of input data into probability using the logistic function. The advantages of logistic regression include simplicity in interpreting results, but it may be limited when analyzing complex relationships between variables and requires an assumption of linearity in relationships. Mathematically, logistic regression models the probability $P(Y=1)$ as a function of X, where Y is the dependent variable, and X is the set of independent variables. The probability is described by the equation:

$$P(Y = 1) = \frac{1}{1+e^{-(\beta_0+\beta_1 X_1+\ldots+\beta_k X_k)}}, \quad (1)$$

where e is the base of the natural logarithm, β_0 is the constant term (intercept), and β_1,..., β_k are the coefficients of the independent variables. This equation allows us to estimate the probability that an observation belongs to class 1, depending on the values of the independent variables.

Support Vector Machine (SVM) algorithm [14] finds a hyperplane in a multidimensional space that best separates different data classes, maximizing the distance between the closest data points (support vectors) of different classes. The advantages of SVM include high accuracy in classification tasks, especially on relatively small datasets, and flexibility through the use of various kernel functions. However, its drawbacks include high computational resource

requirements for large datasets and complexity in interpreting the model. Mathematically, SVM seeks to solve the optimization problem: minimize $\frac{1}{2} || w ||^2$ subject to the constraints that

$$y_i (w \cdot x_i + b) \geq 1 \text{ to all } I, \quad (2)$$

where w is the weight vector of the hyperplane, b is the bias, x_i are the feature vectors, and y_i are the class labels.

The Random Forest algorithm [15] creates an ensemble of decision trees, training each tree on randomly selected subsets of the training dataset and features, which ensures high accuracy and model universality. The advantages of Random Forest include its ability to efficiently handle large datasets with high feature dimensionality and its lower tendency to overfit compared to individual decision trees. However, its drawbacks include relatively high computational resource requirements and the complexity of interpreting the model due to the large number of trees. The mathematical interpretation of Random Forest is based on the principle of "wisdom of the crowd," where the final model decision is determined by voting among the trees for classification tasks or averaging the outputs for regression tasks. More precisely, for classification:

$$Y = mode\{y_1, y_2, \ldots, y_n\}, \quad (3)$$

and for regression:

$$Y = \frac{1}{n} \sum_{i=1}^{n} y_i, \quad (4)$$

where y_i is the prediction of each tree, and Y is the final prediction of the ensemble.

Gradient Boosting [16] is an ensemble machine learning method that improves predictions by sequentially training weak models, typically decision trees, to minimize a loss function. It is characterized by high prediction accuracy and flexibility in parameter tuning, but it can be prone to overfitting if not properly configured and requires more time and computational resources for training compared to other algorithms. Mathematically, Gradient Boosting performs optimization by adaptively reducing the difference between actual and predicted values using gradient descent, where the model update in the m-th iteration is defined as:

$$F_m(x) = F_{m-1}(x) + \alpha_m h_m(x), \quad (5)$$

where $F_{m-1}(x)$ is the prediction at the previous step, $h_m(x)$ is the weak classifier, and α_m is the learning rate.

The k-Nearest Neighbors (KNN) algorithm [17] classifies objects based on the nearest training examples in the feature space, where "k" indicates the number of neighbors considered to determine the class of a new object. The advantages of KNN include ease of implementation and its ability to effectively work with multi-class datasets. However, it requires significant computational resources to store training data and determine neighbors in large datasets, and it is sensitive to irrelevant features and data scaling. Mathematically, the classification of an object x in the KNN algorithm is determined by the majority vote of its neighbors, where each neighbor is weighted according to the inverse distance to x, typically using Euclidean distance:

$$d(x, x_i) = \sqrt{\sum_{j=1}^{m} (x_j - x_{ij})^2}, \quad (6)$$

where x is the point for classification, x_i is a point from the training dataset, and j varies from 1 to m, the number of features. The class of x is determined based on the most frequent class among the k nearest neighbors.

The Decision Tree algorithm [18] builds a predictive model in the form of a tree-like structure by partitioning the dataset into smaller subsets while simultaneously developing the associated decision tree. The advantages of decision trees include ease of interpretation, the ability to handle both numerical and categorical data, and no requirement for data normalization. However, decision trees are prone to overfitting, especially with deep trees, and can be unstable, meaning small changes in data can result in significantly different decision trees. Mathematically, decision trees use the concept of information gain or reduction in uncertainty (entropy) to select the attribute that best splits the dataset into subsets according to the target variable. The information gain for attribute A is calculated as:

$$IG(T, A) = Entropy(T) - \sum_{v \in Values(A)} \frac{|T_v|}{|T|} Entropy(T_v), \quad (7)$$

where T is the training set, $Values(A)$ is the set of all possible values of attribute A, T_v is the subset of T for which attribute A has the value v, and $Entropy(T)$ is the entropy of the training set T.

XGBoost (eXtreme Gradient Boosting) [19] is a highly efficient implementation of the gradient boosting algorithm, which optimizes both linear models and decision trees. The algorithm stands out for its high execution speed, ability to efficiently scale to large datasets, and built-in support for regularization, helping to mitigate overfitting. However, XGBoost can be challenging to tune due to the large number of hyperparameters and requires more time for training compared to simpler models. Mathematically, XGBoost minimizes losses using gradient descent, where the objective function includes both the loss function L and a penalty for model complexity Ω, adding regularization:

$$Obj = \sum_i L(y_i, \hat{y}_i) + \sum_k \Omega(f_k), \quad (8)$$

where y_i are the true labels, $\hat{y}_i$ are the predicted labels, f_k are the functions representing individual trees, and $\Omega(f_k)$ includes terms for regularization, such as the number of leaves and the sum of squares of node weights, thereby reducing the risk of overfitting.

AdaBoost (Adaptive Boosting) [20] is a machine learning algorithm that combines multiple weak classifiers to create a strong classifier, using an iterative approach to correct the errors of previous classifiers by assigning greater weight to observations that are harder to classify. The advantage of AdaBoost is its ability to improve prediction accuracy, ease of implementation, and automatic correction of underperforming classifiers. However, it can be prone to overfitting in the presence of outliers or highly noisy data and requires careful tuning of the number of iterations. Mathematically, AdaBoost adapts the weights of training observations, w_i, by increasing the weights of incorrectly classified observations. At each iteration step t, a classifier h_t is selected to minimize the weighted sum of errors. The final classifier is determined as a weighted sum of these classifiers.

$$H(x) = sign(\sum_{t=1}^{T} \alpha_t h_t(x)), \quad (9)$$

where α_t is the weight assigned to classifier h_t, which depends on its accuracy.

Conclusions drawn from the description of the used classifiers underscore the importance of adapting the model to the specificity of the dataset and the analytical task. The effectiveness of each method depends on the size and quality of the data, the complexity of relationships in the dataset, as well as specific requirements for accuracy and interpretation of results. In the context of disinformation analysis, the choice between the simplicity of interpreting logistic regression and the high accuracy but complexity of tuning XGBoost or SVM may determine the success or failure in identifying fake news. Thus, careful selection and tuning of classifiers are critically important for developing effective tools to combat disinformation in the context of the Russian-Ukrainian war.

3.3 Evaluation metrics

For evaluating the effectiveness of models in classification tasks, key metrics such as precision, recall, F1-score, accuracy, and confusion matrix are utilized [21]. These metrics allow for a deeper analysis of the model's performance, identifying potential weaknesses, and optimizing the model for better results.

Precision is defined as the ratio of the number of true positive results to the total number of results classified as positive by the model. Mathematically, it is represented as:

$$P = \frac{TP}{TP+FP}, \quad (10)$$

where TP — true positives, and FP — false positives.

Recall measures the model's ability to identify all actual positive cases in the dataset. It is defined as the ratio of the number of correctly identified positive results to the sum of correctly identified positive results and instances that are actually positive but were missed by the model. The formula for calculation is:

$$R = \frac{TP}{TP+FN}, \quad (11)$$

where FN — false negatives.

The F1 Score is the harmonic mean between precision and recall, providing a balance between these two metrics. This is particularly useful in situations where class imbalances may cause biases in one metric over the other. F1 is defined as:

$$F1 = 2 \cdot \frac{P \cdot R}{P+R}. \quad (12)$$

Accuracy measures the percentage of cases correctly classified by the model and is defined by the formula:

$$A = \frac{TP+TN}{TP+TN+FP+F}, \quad (13)$$

where TN — true negatives.

The Confusion Matrix provides a visualization of classification results by representing the counts of TP, TN, FP, and FN in the form of a matrix. This allows us not only to determine the accuracy of the model but also to understand the types of errors made by the model.

3.4 Model training

The training procedure (Figure 4) on the training dataset is a fundamental step in the development of effective machine learning algorithms for text data classification. In this context, the use of datasets "news_data.csv" and "data_set_4.csv" for training and testing models, respectively, provides a valuable foundation for misinformation analysis. The initialization of the procedure begins with the import and preprocessing of data, including the removal of records without textual content, ensuring data cleanliness for subsequent processing stages.

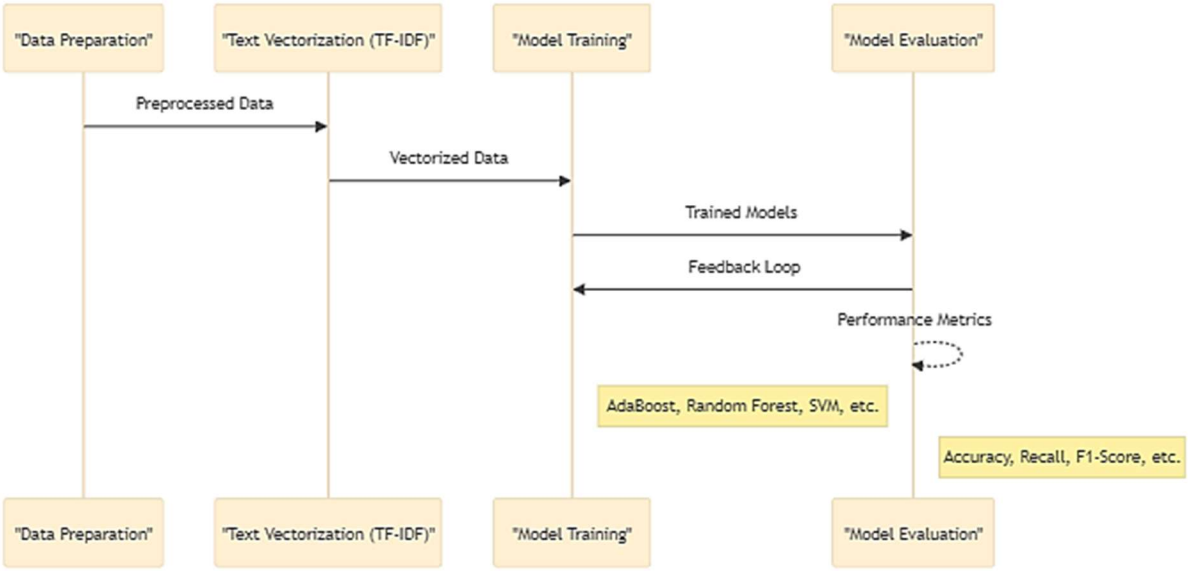

Figure 4: Model Training Process

Text vectorization using TF-IDF (Term Frequency-Inverse Document Frequency) is a key step in data preparation, as it transforms textual data into a numerical format, making them suitable for processing by machine learning models. This method considers not only the frequency of words in the text but also their uniqueness through the inverse frequency of documents, allowing the model to better identify important features in the text.

After preparing the vectorized training and testing data, the next stage involves training different models on the training dataset. This process requires the application of machine learning algorithms to the training dataset to form a model capable of effectively classifying text based on learned features. Each model adjusts its parameters to minimize errors on the training set while simultaneously ensuring the ability to generalize to new data.

Evaluation of the effectiveness of each trained model on the test dataset is crucial for determining its suitability and efficiency in classification tasks. The use of evaluation metrics such as accuracy, recall, F1-score, and overall accuracy allows for deep analysis and comparison of model performance. The evaluation results can be visualized for better understanding of model effectiveness, and the best-performing model can be selected for further use or saved for future analysis.

Thus, the model training procedure on the training dataset using pre-processed and vectorized text data is fundamental for developing reliable machine learning tools capable of effectively classifying and analyzing text for misinformation.

4. Research results

Further, we will discuss the implementation and evaluation of the effectiveness of various machine learning models in the task of classifying misinformation data in the Ukrainian media space, contextualized in the context of Russia's full-scale intervention. By analyzing a wide range of algorithms, we aim to identify the most effective methods for accurate detection and differentiation of misinformation incidents. This analysis is based on a comprehensive comparison of overall classification accuracy as well as specific metrics such as precision, recall, and F1-score for each class. The implementation was done in the Python programming language utilizing libraries for text analysis.

The logistic regression model showed (Figure 5) an overall classification accuracy of 86.4% on the test sample of 10,735 examples, demonstrating high effectiveness in identifying true values with an F1-score of 0.92 for the "True" class. However, the model was less accurate in identifying the "False" class, with a prediction accuracy of 0.96 and a low recall score of 0.44, indicating a significant number of false negatives, as reflected in the confusion matrix with 1,407 misclassified examples.

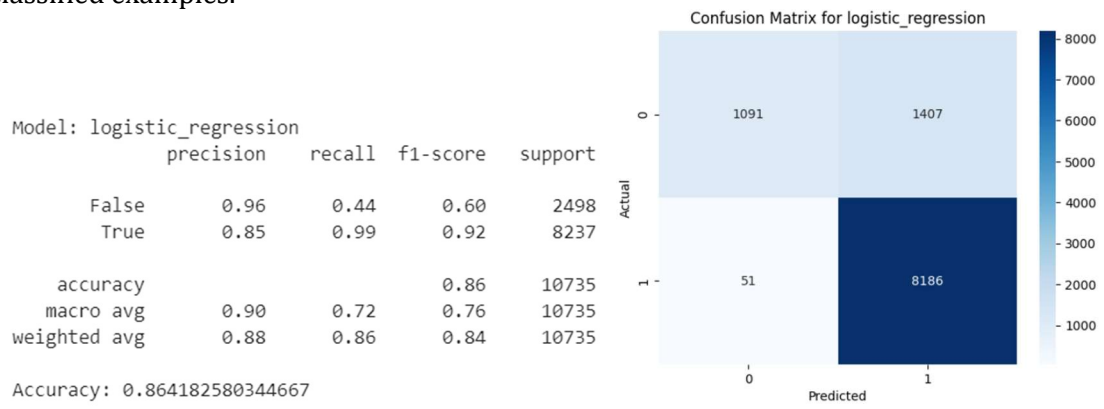

Figure 5: LogisticRegression Results

The SVM model demonstrated (Figure 6) high overall classification accuracy of 93.6% for the test sample, indicating its effectiveness in distinguishing between the "True" and "False" classes. Specific accuracy and recall metrics for the "False" class were 0.97 and 0.75, respectively, demonstrating the model's ability to identify negative cases well, albeit with some errors. At the same time, high metrics for the "True" class with precision of 0.93 and recall of 0.99 indicate minimal false negative classifications, confirmed by a low number of errors in the confusion matrix (618 for "False" and 68 for "True").

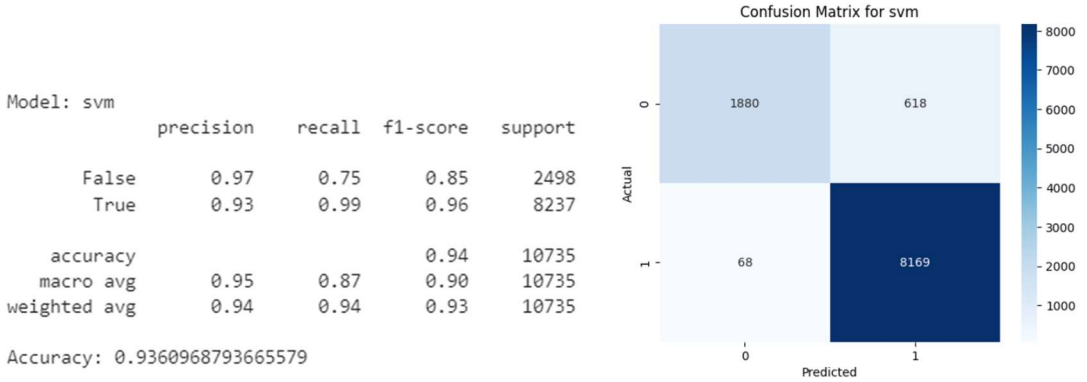

Figure 6: SVM Results

The Random Forest model demonstrated (Figure 7) high accuracy in classification with an overall accuracy of 95.3% on the test dataset, indicating the model's high effectiveness in recognizing both classes. For the "False" class, the model showed high accuracy (0.98) and a relatively high recall of 0.81, indicating the model's ability to effectively identify negative cases with a moderate number of errors. Conversely, extremely high accuracy (0.95) and almost perfect

recall (1.00) for the "True" class highlight the minimal number of false negative results, corroborated by the low number of errors in the confusion matrix.

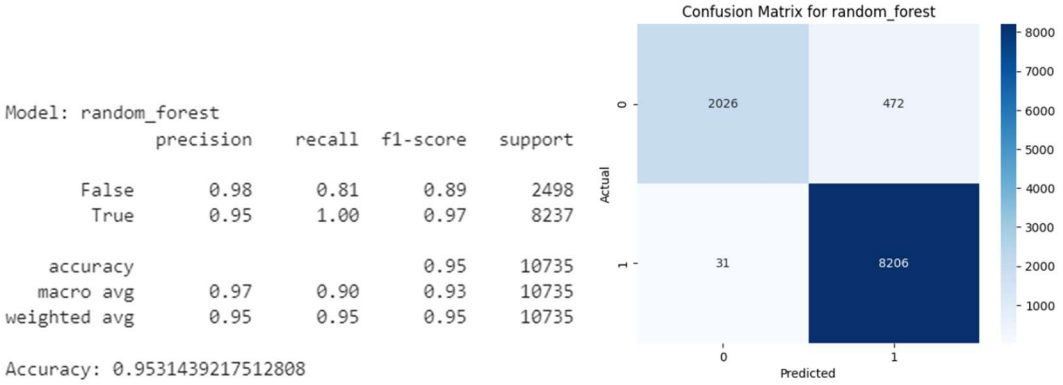

Figure 7: RandomForestClassifier Results

The Gradient Boosting model achieved (Figure 8) an overall classification accuracy of 87.3% on the test dataset, indicating the model's good ability to distinguish between the "True" and "False" classes. The model's precision for the "False" class was high (0.94), but the recall was only 0.48, indicating a significant number of type II errors, where negative cases are often misclassified as positive. Conversely, for the "True" class, the model showed impressive classification ability with a precision of 0.86 and a recall of 0.99, demonstrating its high effectiveness in identifying true positive cases with minimal errors.

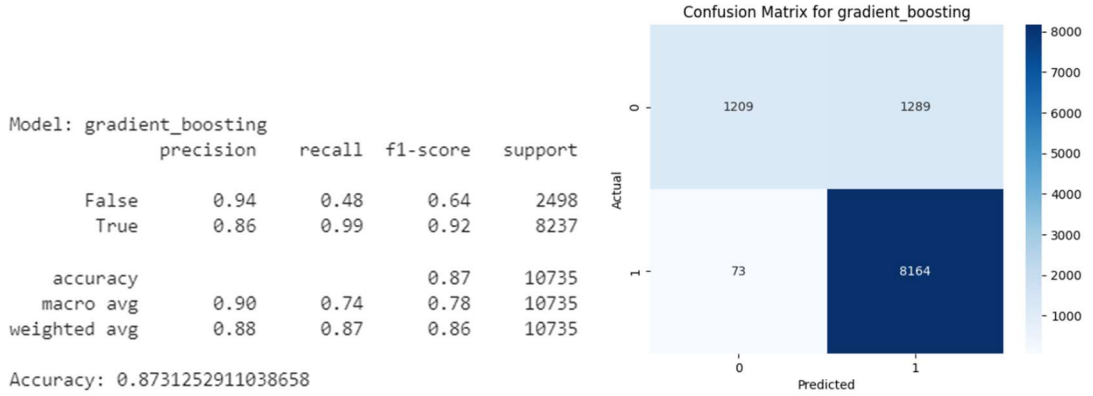

Figure 8: Gradient Boosting Results

The KNN (K-Nearest Neighbors) model demonstrated (Figure 9) an overall classification accuracy of 87.6% on the test dataset, highlighting its ability to effectively classify data. Although the model showed high precision (0.91) for the "False" class, the recall was only 0.51, indicating difficulties in identifying all negative cases. Conversely, for the "True" class, the model exhibited excellent precision (0.87) and a high recall (0.99), indicating its ability to identify positive cases with high confidence, albeit with some false positive errors, as seen in the confusion matrix.

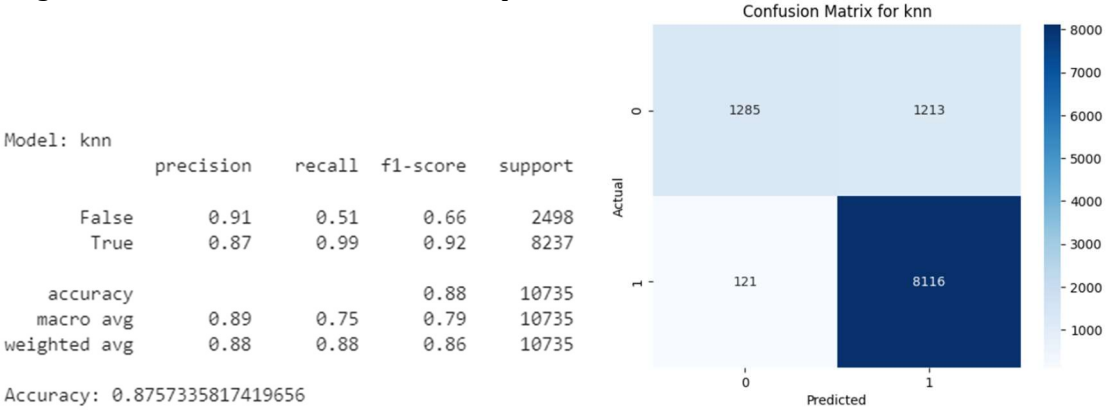

Figure 9: KNN Results

The Decision Tree model achieved (Figure 10) a high level of classification accuracy, 91.4%, on the test dataset, confirming its effectiveness in classifying data into "True" and "False" classes. For the "False" class, the model showed relatively high precision (0.81) and recall (0.83), indicating a balanced ability to identify negative cases with a relatively small number of errors. Meanwhile, for the "True" class, the model provided impressive precision (0.95) and recall (0.94), demonstrating its strong capabilities in identifying positive cases with minimal false negatives, as reflected in the confusion matrix.

Model: decision_tree

	precision	recall	f1-score	support
False	0.81	0.83	0.82	2498
True	0.95	0.94	0.94	8237
accuracy			0.91	10735
macro avg	0.88	0.89	0.88	10735
weighted avg	0.92	0.91	0.91	10735

Accuracy: 0.9144853283651607

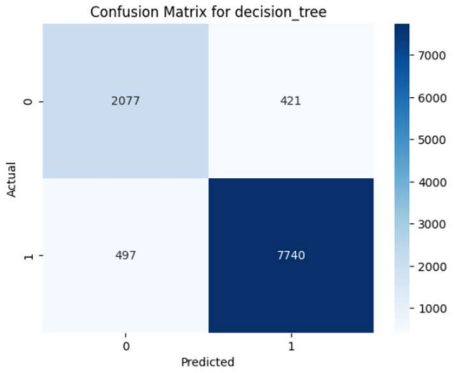

Figure 10: DecisionTreeClassifier Results

The XGBoost model demonstrated (Figure 11) an overall accuracy of 89.2% on the test dataset, indicating its ability to effectively classify the given dataset. The precision and recall metrics for the "False" class were 0.89 and 0.61, respectively, indicating the model's higher ability to correctly identify negative cases, albeit with some errors. Meanwhile, for the "True" class, the model showed high precision and recall (both 0.89 and 0.98), demonstrating excellent ability to accurately identify positive cases with minimal errors, as reflected in its high F1-score.

Model: xgboost

	precision	recall	f1-score	support
False	0.89	0.61	0.73	2498
True	0.89	0.98	0.93	8237
accuracy			0.89	10735
macro avg	0.89	0.80	0.83	10735
weighted avg	0.89	0.89	0.88	10735

Accuracy: 0.8919422449930136

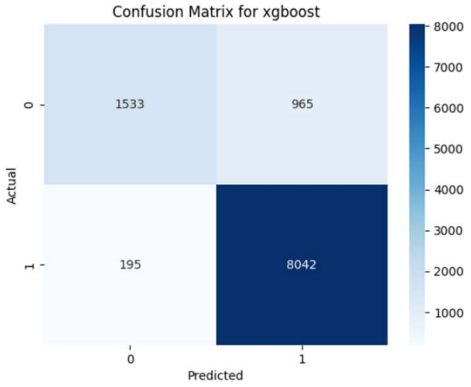

Figure 11: XGBoost Results

The AdaBoost model exhibited (Figure 12) an overall classification accuracy of 86.9% on the test dataset, indicating its effectiveness in recognizing data, albeit with some limitations. For the "False" class, the model achieved a precision of 0.88 with a recall of 0.50, indicating a relatively low ability to identify all negative cases. On the other hand, high precision (0.87) and recall (0.98) for the "True" class underscore the model's strong ability to detect positive cases, albeit with few errors, as reflected in the confusion matrix.

Model: adaboost

	precision	recall	f1-score	support
False	0.88	0.50	0.64	2498
True	0.87	0.98	0.92	8237
accuracy			0.87	10735
macro avg	0.88	0.74	0.78	10735
weighted avg	0.87	0.87	0.86	10735

Accuracy: 0.8692128551467163

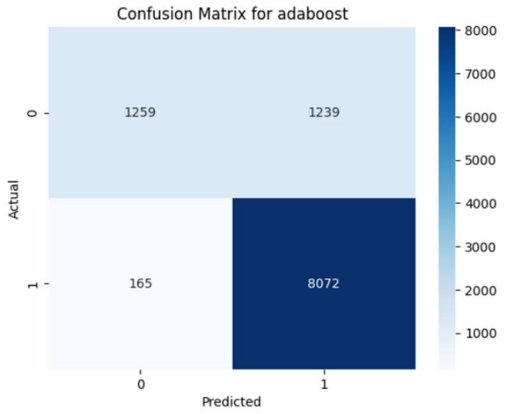

Figure 12: AdaBoostClassifier Results

Therefore, analyzing the results of applying various machine learning models to classify disinformation data in the Ukrainian media space after the onset of Russia's full-scale invasion, it can be noted that the Random Forest model proved to be the most effective with an accuracy of 95.3%, highlighting its ability to accurately detect and differentiate disinformation incidents. At the same time, models such as AdaBoost and logistic regression showed lower overall accuracy, which may indicate their limitations in identifying subtle cases of disinformation or more subjective aspects of information operations. This underscores the importance of choosing the appropriate model for the task of analyzing disinformation, where Random Forest may be more suitable for deep understanding and detecting complex patterns of disinformation in the context of information warfare.

5. Conclusion

The analysis of the effectiveness of various machine learning models applied to the task of classifying news headlines for misinformation in the Ukrainian media space demonstrates significant variations in accuracy, recall, and F1-score among the models. The considered algorithms, including logistic regression, SVM, random forest, gradient boosting, KNN, decision tree, XGBoost, and AdaBoost, showed different levels of performance in addressing the task.

The random forest model emerged as the most effective, achieving an overall accuracy of 95.3%, indicating its high capability to recognize and distinguish true and false messages. This is supported by high precision scores for both the "False" class (0.98) and the "True" class (0.95), as well as significant recall scores for both classes (0.81 for "False" and 1.00 for "True"), demonstrating its effectiveness in minimizing both false positives and false negatives.

These results underscore the importance of choosing the appropriate model for a specific misinformation analysis task. The model choice not only affects the overall classification accuracy but also the model's ability to minimize false positives or false negatives, which is crucial for developing effective tools to combat misinformation. Particularly, the random forest model, which demonstrated the best performance, can be recommended as the optimal choice for similar tasks, providing a high level of accuracy and the ability to effectively distinguish between true and false messages.

Further scientific research in the field of identification and analysis of misinformation in the Ukrainian media space requires deeper development and improvement of machine learning algorithms, with a particular focus on enhancing their ability to recognize subtle and complex forms of misinformation. The results of our study show that the random forest model, with an accuracy of 95.3%, proved to be the most effective, but there is potential for improvement, especially in accurately distinguishing between true and false messages. In the future, researchers may focus on developing hybrid models that combine the advantages of multiple algorithms, including deep learning and neural networks, to ensure greater adaptability and accuracy in different informational contexts. Additionally, an important direction will be the development of methods that allow models to better understand the semantic context and emotional tone of texts, which can significantly improve their ability to identify hidden

misinformation. Implementing such approaches will require not only technological innovations but also a deeper understanding of linguistic nuances and cultural-historical contexts on which misinformation is based.

References

[1] Ashford, J.R. (2024). Detecting Anti-vaccine Content on Twitter using Multiple Message-Based Network Representations. arXiv preprint arXiv:2402.18335.

[2] Donabauer, G., & Kruschwitz, U. (2024). Challenges in Pre-Training Graph Neural Networks for Context-Based Fake News Detection: An Evaluation of Current Strategies and Resource Limitations. arXiv preprint arXiv:2402.18179.

[3] Kaur, S., & Ranjan, S. (2024). Comparative Analysis of Supervised and Unsupervised Machine Learning Algorithms for Fake News Detection: Performance, Efficiency, and Robustness. ResearchGate.

[4] Vahdat-Nejad, H., Akbari, M. G., Salmani, F., Azizi, F., & Nili-Sani, H. R. (2023). Russia-Ukraine war: Modeling and Clustering the Sentiments Trends of Various Countries. arXiv preprint arXiv:2301.00604.

[5] Haq, E. U., Tyson, G., Lee, L. H., Braud, T., & Hui, P. (2022). Twitter dataset for 2022 russo-ukrainian crisis. arXiv preprint arXiv:2203.02955.

[6] Zia, H. B., Haq, E. U., Castro, I., Hui, P., & Tyson, G. (2023). An Analysis of Twitter Discourse on the War Between Russia and Ukraine. arXiv preprint arXiv:2306.11390.

[7] Daria, S. (2023). When a Language Question Is at Stake. A Revisited Approach to Label Sensitive Content. arXiv preprint arXiv:2311.10514.

[8] Racek, D., Davidson, B. I., Thurner, P. W., & Kauermann, G. (2023). The Politics of Language Choice: How the Russian-Ukrainian War Influences Ukrainians' Language Use on Twitter. arXiv preprint arXiv:2305.02770.

[9] Zhu, Y., Haq, E. U., Lee, L. H., Tyson, G., & Hui, P. (2022). A reddit dataset for the russo-ukrainian conflict in 2022. arXiv preprint arXiv:2206.05107.

[10] Padalko, H., Chomko, V., & Chumachenko, D. (2023). Misinformation Detection in Political News using BERT Model. Proceedings of the 3rd International Workshop of IT-professionals on Artificial Intelligence (ProfIT AI 2023) Waterloo, Canada, November 20-22, 2023. 117-127

[11] Ukrainian news. (n.d.-a). Kaggle: Your Machine Learning and Data Science Community. https://www.kaggle.com/datasets/zepopo/ukrainian-fake-and-true-news/data?select=news_data.csv

[12] Lipyanina, H., Maksymovych, V., Sachenko, A., Lendyuk, T., Fomenko, A., & Kit, I. (2020, August). Assessing the investment risk of virtual IT company based on machine learning. In International Conference on Data Stream Mining and Processing (pp. 167-187). Cham: Springer International Publishing.

[13] Lipyanina, H., Sachenko, S., Lendyuk, T., Brych, V., Yatskiv, V., & Osolinskiy, O. (2021, January). Method of detecting a fictitious company on the machine learning base. In International Conference on Computer Science, Engineering and Education Applications (pp. 138-146). Cham: Springer International Publishing.

[14] Kalogridis, I. (2024). Robust and adaptive functional logistic regression. Computational Statistics & Data Analysis, 192, 107905.

[15] Çakir, M., Yilmaz, M., Oral, M. A., Kazanci, H. Ö., & Oral, O. (2023). Accuracy assessment of RFerns, NB, SVM, and kNN machine learning classifiers in aquaculture. Journal of King Saud University-Science, 35(6), 102754.

[16] Sun, Z., Wang, G., Li, P., Wang, H., Zhang, M., & Liang, X. (2024). An improved random forest based on the classification accuracy and correlation measurement of decision trees. Expert Systems with Applications, 237, 121549.

[17] Wang, C., Xiao, W., & Liu, J. (2023). Developing an improved extreme gradient boosting model for predicting the international roughness index of rigid pavement. Construction and Building Materials, 408, 133523.

[18] Çakir, M., Yilmaz, M., Oral, M. A., Kazanci, H. Ö., & Oral, O. (2023). Accuracy assessment of RFerns, NB, SVM, and kNN machine learning classifiers in aquaculture. Journal of King Saud University-Science, 35(6), 102754.
[19] Gao, B., Zhou, Q., & Deng, Y. (2024). HIE-EDT: Hierarchical interval estimation-based evidential decision tree. Pattern Recognition, 146, 110040.
[20] Niazkar, M., Menapace, A., Brentan, B., Piraei, R., Jimenez, D., Dhawan, P., & Righetti, M. (2024). Applications of XGBoost in water resources engineering: A systematic literature review (Dec 2018–May 2023). Environmental Modelling & Software, 105971.
[21] Xing, H. J., Liu, W. T., & Wang, X. Z. (2024). Bounded exponential loss function based AdaBoost ensemble of OCSVMs. Pattern Recognition, 148, 110191.