STUDIA UNIV. BABEŞ–BOLYAI, INFORMATICA, Volume **LXXI**, Number 1-2, 2026
DOI: 10.24193/subbi.2026.07

FEATURE SELECTION FOR CHARACTER N-GRAMS IN ROMANIAN AUTHORSHIP ATTRIBUTION

LUPŞA DANA, LUPŞA RADU

Abstract. Character N-gram representations are widely used in authorship attribution but often generate extremely high-dimensional feature spaces, making feature selection an important design decision. This study investigates the impact of eight feature selection methods on Romanian authorship attribution using four classifiers and character 3-, 4-, and 5-gram representations. The best overall result was achieved by a Random Forest classifier using F-test filtering with only 500 3-gram features. Despite using substantially fewer features, the model achieved performance comparable to the best results obtained by models trained on the full feature set and exceeded it by a small margin. These findings highlight the potential of feature selection to develop efficient systems without sacrificing accuracy.

1. Introduction

Authorship attribution (**AA**) is the process of automatically identifying the author of a text based on its stylistic and linguistic features. The field has attracted considerable attention due to its importance in a variety of applications, including plagiarism detection, digital humanities (e.g., resolving historical literary disputes) or forensic linguistics (e.g., analyzing threatening notes or ransom demands) [6, 9, 12, 20, 22]. By identifying patterns that reflect an individual's writing habits, authorship attribution systems can assist in resolving questions of authorship in both academic and real-world settings.

In authorship attribution, **N-grams** are one of the most effective and widely adopted feature types due to their ability to capture stylistic and syntactic patterns that characterize an author's writing style [3, 13, 22, 23]. Commonly

Received by the editors: 8 September 2026.

2010 *Mathematics Subject Classification.* 68T50.

1998 *CR Categories and Descriptors.* [**Artificial Intelligence**]: Natural Language Processing – *Text Analysis*; [**Artificial Intelligence**]: Learning – *Induction.*

Key words and phrases. Authorship Attribution, Machine Learning, Feature Selection.

© Studia UBB Informatica. Published by Babeş-Bolyai University

This work is licensed under a Creative Commons Attribution-NonCommercial-NoDerivatives 4.0 International Licence.

used representations include character N-grams, word N-grams, and part-of-speech (POS) N-grams. Character N-grams in particular have demonstrated remarkable effectiveness and robustness across genres, topics, and languages, making them one of the most useful feature type.

Despite their effectiveness, N-gram representations typically result in very high-dimensional feature spaces, especially for larger values of N. As a consequence, the computational cost of training and inference increases, while classification performance may suffer [8, 11]. Feature filtering constitutes an important preprocessing step, enabling the selection of the most informative features while reducing dimensionality.

Feature filtering, and feature selection in general, has been extensively studied in natural language processing and machine learning. Investigating feature selection is valuable not only because it has the potential to improve the accuracy of text authorship attribution, but also because it can enhance model interpretability, by identifying linguistic elements that are most discriminative [5, 11]. Unlike deep learning, which works well when massive datasets are available and hide features in the internal representation, approaches that use feature selection help identify the linguistic patterns that characterize individual authors and reduce computational complexity. Thus, feature selection serves both as an optimization technique and as an analytical tool in stylometric research.

This work investigates feature selection methods for reducing feature space dimensionality when using character N-gram features while maintaining classification performance. Experiments were conducted using four machine learning models and several feature subset sizes. We examine the impact of feature subset dimensionality on classification performance. We compare the scores obtained by using selection techniques against the results obtained using the full feature sets and highlight cases in which reduced feature sets outperformed the full-feature representation. In addition, the study examines the relationship between feature subset size and classification effectiveness, showing that, for some classifiers, the best results are achieved with feature sets containing only 500 or 1000 features.

This research extends previous studies on Romanian-language authorship attribution based on the ROST dataset [2, 1, 3, 15, 16]. To the best of our knowledge, it represents the first study of feature selection for Romanian authorship attribution. Although previous studies [18] have incorporated feature filtering, they did not investigate multiple filtering approaches or evaluate their effectiveness across several machine learning classifiers.

2. Feature selection

Feature selection seeks to identify a subset of relevant features from the original feature space while eliminating noisy and irrelevant features that may negatively affect model performance [4, 5, 7, 17, 19]. Filtering techniques are particularly valuable in applications characterized by high-dimensional data and relatively small sample sizes.

In addition to their use in tasks related to natural language processing, feature selection methods have been successfully applied in numerous other domains, such as medicine, bioinformatics, genomics, image processing, and finance. Research in feature filtering and selection for authorship attribution is important because the initial "style" representation can involve hundreds of thousands of features (e.g., 5-grams in our study), many of which are expected to be redundant, irrelevant, or noisy and may negatively affect classification performance.

Feature selection methods are commonly divided into three major categories [4, 5, 24]: filter methods, which assess the relevance of features using statistical properties of the data independently of any learning algorithm; wrapper methods, which evaluate feature subsets in relation with the performance of a classifier; and embedded methods, which integrate feature selection into the model training process.

In this study, we focus primarily on filter-based approaches due to their computational efficiency and scalability to high-dimensional text representations. The evaluated methods include Chi-Square (Chi2), F-test, Kruskal-Wallis, Mutual Information (MI), Variance Filtering, and ReliefF. In addition, we consider two model-based approaches that estimate feature importance using trained classifiers: Random Forest feature importance (RF-based) and Logistic Regression with L2 regularization (LR-L2). The selected methods represent a diverse set of statistical, similarity-based, and model-driven feature evaluation strategies, allowing a comprehensive comparison of their effectiveness for Romanian authorship attribution.

The statistical methods differ in the criteria used to assess feature relevance. Chi-Square measures the degree of dependence between individual features and class labels, while Mutual Information quantifies the amount of information shared between a feature and the target variable. The F-test and Kruskal-Wallis test assess the discriminative ability of features across classes using parametric and non-parametric statistical criteria, respectively. The F-test compares class means under the assumptions of normality and homogeneity of variance, whereas the Kruskal-Wallis test converts data to ranks and compares mean ranks across groups and does not require distributional assumptions.

Variance Filtering removes features with little variation across documents, under the assumption that a feature that does not change across samples cannot help a machine learning model distinguish between different classes or categories.

Relief [24] is a feature selection algorithm that estimates feature relevance based on nearest-neighbor comparisons. For each instance in the dataset, Relief identifies neighboring samples from the same class (Near Hits) and from different classes (Near Misses). Unlike the original Relief algorithm, ReliefF supports multi-class classification problems and exhibits greater robustness to noisy and incomplete data.

To complement the filter-based methods, we also evaluated two model-driven feature selection approaches. The first, referred to as RF-based, derives feature importance scores from a Random Forest classifier, where importance is estimated according to the cumulative reduction in Gini impurity produced by splits involving each feature. The RF model was configured with 100 trees and a fixed random seed (random_state = 42) to ensure reproducibility. The second approach uses Logistic Regression with L2 regularization and ranks features based on the absolute values of the coefficients learned by the model. The regularization parameter was set to $C = 10.0$, reducing the strength of regularization and allowing the model greater flexibility to assign larger weights to discriminative features. To ensure reproducibility, the random seed was also fixed at 42.

3. Methodology and Implementation

3.1. **Classification Algorithms.** To analyze the impact of feature selection methods we selected four learning algorithms based on distinct theoretical foundations. K Nearest Neighbors (kNN) is a distance-based model: it classifies a text according to its nearest neighbors. Logistic Regression (LR) is a probabilistic model; it models author attribution by computing conditional class probabilities. Random Forests (RF) combine an ensemble of randomized decision trees and aggregates their predictions. Support Vector Machines (SVM) is a margin-based model: it optimizes a maximum-margin separating decision boundary between classes in a high-dimensional feature space. We selected models from distinct foundational families to see if feature selection can change performance across diverse learning paradigms. The learning algorithms were implemented via the `scikit-learn` library and the parameter configuration followed those reported in [15], ensuring consistency with prior experiments performed on the same dataset. The Support Vector Machine uses a linear kernel and default hyperparameter tuning. Logistic Regression uses the lbfgs solver with L2 regularization. K Nearest Neighbors uses the

Minkowski distance metric, and its neighborhood size is varied across five values $k \in \{3, 5, 7, 9, 11\}$. Random Forest uses 100 estimators and five different random seeds (7, 17, 42, 67, and 101).

3.2. **The dataset.** The ROST (ROmanian Stories and Other Texts) corpus is one of the most important publicly available resources for Romanian-language authorship attribution research. The corpus contains 400 documents written by 10 different authors and the distribution of texts across authors is uneven; the number of documents associated with each author ranges from 27 to 60, resulting in a substantial class imbalance.

While English authorship attribution benefits from massive, diverse benchmarks, Romanian is a lower-resourced language with far fewer standardized datasets in the domain of natural language processing. ROST can be used as a common ground for tracking the evolution of AA techniques, being used in research focused on evaluating linguistic features, as well as investigating the performance of machine learning and deep learning models [2, 3, 15, 16, 18].

Following the work in [15], we adopted the normalization used there. This involved standardizing diacritics, unifying punctuation, whitespace regularization and substituting all digits with a dedicated placeholder symbol (@). Furthermore, spaces and tabs were replaced with the underscore character, and paragraph boundaries were notated with the dollar sign ($), allowing the N-gram extraction process to capture transitions between lexical items and between paragraphs, treating them as separate tokens.

3.3. **Evaluation.** In our experiments, for each configuration that we evaluated, the dataset was partitioned into 80% for model training and 20% for performance testing, using `scikit-learn`'s `train_test_split` function, with the following seeds: $7, 17, 42, 67, 101$. To ensure reliable performance estimates, the data partitioning was repeated five times using distinct random seeds. For each resulting split and subsequent evaluation, five runs were executed.

Performance was evaluated using the next metrics: accuracy, F1 and recall. Accuracy measures the proportion of correctly classified instances. Recall quantifies the proportion of texts correctly attributed to each author. Precision quantifies the proportion of predicted author assignments that are correct. The F1-score combines both recall and precision into a single measure, ensuring that strong performance requires both accurate identification of an author's texts and reliable exclusion of texts written by others.

Since the ROST corpus is imbalanced, overall performance indicators may be disproportionately influenced by majority classes, leading to an overestimation of performance for authors with a large number of available documents. To address this effect, all reported metrics are macro-averaged, giving equal weight

to each author regardless of corpus size. Macro-averaged accuracy (Acc) is used as the primary evaluation measure, while macro-F1 (F1) and macro-recall (Recall) are reported as complementary indicators.

N-gram size	split 1	split 2	split 3	split 4	split 5
N=2	1807	1749	1758	1738	1728
N=3	14875	14577	14543	14508	14637
N=4	78037	77021	76077	77271	77510
N=5	269117	266154	262406	268676	267794

TABLE 1. Feature set dimensions for each of the 5 used train-test split, and for character sequence length N varying from 2 to 5.

3.4. **Features.** A character N-gram feature is defined as a contiguous sequence of N characters. Character N-grams are extracted from the text using a sliding window of length N, which is moved one character at a time, resulting in overlapping windows. Each document is then represented as a numerical feature vector, where the value associated with each N-gram is computed using the Term Frequency-Inverse Document Frequency (TF-IDF) weighting scheme.

To evaluate the efficacy of feature selection on high-dimensional representations, our experiments focus on character N-grams with order N being in $\{3, 4, 5\}$. As detailed in Table 1, 2-gram representations yield at most 1807 unique features in our training data, whereas setting $N \geq 3$ expands the feature space significantly to at least 14508 distinct sequences in our training. Our core objective is to investigate whether aggressive feature selection can match or surpass baseline performance while dramatically reducing dimensionality. Consequently, we focus on 3-, 4-, and 5-gram feature sets, each exceeding 10000 original features.

3.5. **Feature set sizes.** Our research analyses the results obtained when the sizes of feature subset were chosen to significantly reduce the dimensionality of the feature space —an approach we name *aggressive feature selection* (Section 4). We evaluate the model's behavior by reducing the feature space to 50, 100, 500, and 1,000 selected features. A second aspect of our study (Section 5) examines the effect of feature set size on classification performance. For this analysis, we used feature set sizes of 50, 100, 500, 1000, 1500, 2000, 2500, 3000.

The sizes of the selected feature sets are chosen to align with the growth of the full feature space, which increases by roughly an order of magnitude from N = 3 (around 14500 features) to N = 5 (around 269000 features).

In the case of aggressive dimensionality reduction, the sets of 50 and 100 features represent less than 1% of the original 3-gram feature space, which

comprises 14500 features. These correspond to the most severe reductions. A set of 500 features represents approximately 3.4% of the original space, while 1000 features represent approximately 6.9%. Furthermore, these percentages are even lower for 4-gram and 5-gram feature spaces.

Prior studies on word N-gram feature selection [8, 10] have generally considered fewer than 3000 selected features. [8] reported that evaluation measures peaked between 500 and 1000 features for the best-performing selection metric, with performance beginning to decline around 2000 features. The same study noted that "if features are selected well, most information should be contained in the initial features selected". Furthermore, the upper bound of 2000 features was motivated in [8] by findings in [25], which indicated that evaluation measures decrease for larger values. Consequently, we selected a range extending from 50 to 3000 features to try to capture the potential rising, plateau, or decline regions of the performance curve.

4. Aggressive feature selection. Results and discussion

To examine the impact of severe dimensionality reduction, we use the term aggressive feature selection to refer to experimental settings in which only a very small subset of features is retained. Specifically, the selected feature set sizes are limited to 50, 100, 500, and 1000 features, all well below the dimensionality of the original feature space. This experimental setup allows us to examine the behavior of different classifiers when operating with significantly reduced feature spaces.

4.1. **Baseline.**

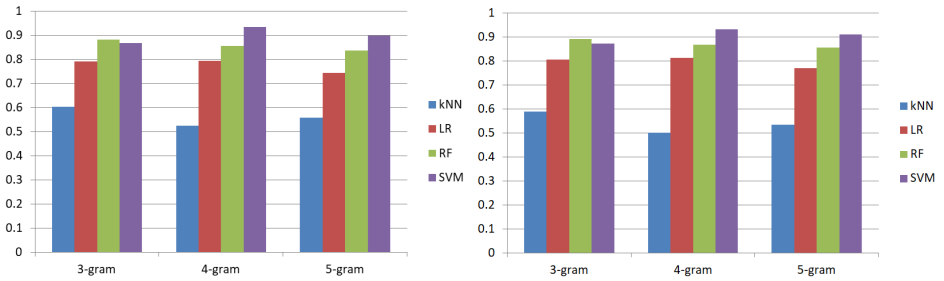
We compare the scores obtained by using selection techniques against the results obtained using the full feature set, in order to determine if feature selection can achieve better performance.

The average performance measured with Acc and F1 metrics over the full feature set and for the four classifiers we use in these experiments is illustrated in Figure 1. RF is the top-performing model for 3-gram sequences, and SVM obtain better results in case of 4-gram and 5-gram sequences.

4.2. **Overview of the results.**

The scores achieved by each classification algorithm across all feature selection methods and for N-gram orders ranging from 3 to 5 are shown in Figure 2. Within each panel (fixed N-gram order and feature selection method), the four bar groups correspond to kNN, LR, RF, and SVM classification models. Values shown correspond to the averaged Acc scores obtained across multiple experimental runs, comprising five train-test splits and five runs per split, using 50, 100, 500, and 1000 selected features.

The overall results show that only kNN and RF benefit from feature selection, achieving better performance for most feature selection models than the



(A) (B)

FIGURE 1. Averaged (A) Acc and (B) F1 values across different N-gram sizes ($N \in \{3, 4, 5\}$) for four classification methods. Values shown represent the average of the Acc and F1 respectively, scores computed over multiple experimental runs.

all-feature model, which is the baseline in our approach. In contrast, LR and SVM perform significantly better when trained on the full feature set.

4.3. **Top scorer.** Table 2 summarizes the top 10 configurations for each N gram order ($N \in \{3, 4, 5\}$), ranked by average macro-accuracy (Acc). Each configuration is defined by a classifier, a feature selection method, and a feature subset size. Experiments were performed with four feature set sizes. To highlight the impact of feature selection, the table also includes the results obtained by RF and SVM using the complete feature set. For each configuration, the reported Acc, F1, and Recall values correspond to averages computed over multiple runs. The table additionally reports the standard deviation of Acc and the maximum Acc. The maximum value represents the highest score obtained across all runs, for any of the data splits and for any of the configurations. The results are presented in descending order based on the average Acc score.

4.4. **Discussion of the results.** The results (see Table 2) indicate a few notable trends. Random Forest (RF) is the classifier used in all top 10 ranked configurations with selected features, proving superior performance across the evaluated settings. Of all the feature selection methods considered, variance and ReliefF filtering do not appear in any of the top 10 ranked results, suggesting that they are less effective than the other feature selection techniques.

When all available features are used, the highest Acc average is 0.93443, achieved by SVM with N = 4. This baseline was matched or slightly exceeded by four feature-selection configurations: three using N = 3 and 500 selected features, based on the F-test, RF-based, and Kruskal-Wallis selection methods, and one using 1000 selected features with Chi2 filtering. These configurations

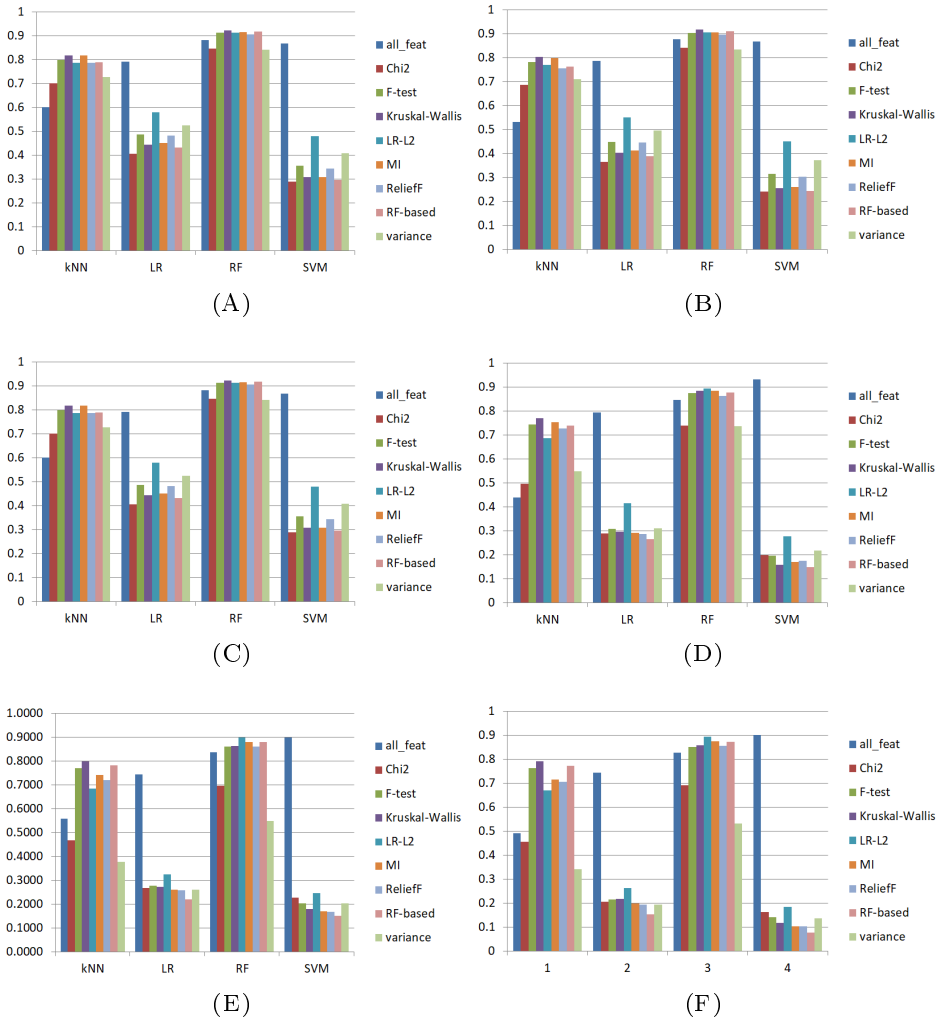


FIGURE 2. Averaged Acc (first column) and F1 (second column) scores obtained for different N-gram sizes ($N \in \{3, 4, 5\}$, on lines 3, 4, 5 respectively), across all feature selection methods and the four classification algorithms: kNN, LR, RF and SVM. The reported values are the averages of Acc and F1 scores computed over the experimental runs for each configuration. The results show that only kNN and RF benefit from feature selection, both outperforming, in most cases, the corresponding models that use all features.

	Model	Feature Selection	No Feat.	Acc avg	Acc max	Acc std	F1 avg	Recall avg
	N=3			**top 10**				
1	RF	F-test	500	0.94463	1	0.02773	0.94025	0.94463
2	RF	Chi2	1000	0.93834	0.98571	0.02691	0.93827	0.93834
3	RF	RF-based	500	0.93769	0.99286	0.02617	0.93728	0.93769
4	RF	Kruskal-Wallis	500	0.93704	0.97857	0.02382	0.93728	0.93704
5	RF	RF-based	1000	0.93428	0.97857	0.02757	0.92988	0.93428
6	RF	Kruskal-Wallis	100	0.93373	0.97468	0.02657	0.92840	0.93373
7	RF	F-test	1000	0.93331	0.98571	0.03037	0.93482	0.93331
8	RF	Kruskal-Wallis	1000	0.93116	0.98571	0.02750	0.93284	0.93116
9	RF	LR-L2	1000	0.92962	0.97857	0.02684	0.93037	0.92962
10	RF	LR-L2	500	0.92884	0.96571	0.01887	0.92741	0.92884
	...							
28	RF	all-feat		0.88327	0.93939	0.0383	0.89284	0.88327
29	SVM	all-feat		0.86886	0.91917	0.0589	0.87407	0.86886
	N=4			**top 10**				
1	SVM	all-feat		0.93443	0.96829	0.03777	0.93333	0.93443
2	RF	Kruskal-Wallis	500	0.92011	0.98571	0.03407	0.92296	0.92011
3	RF	F-test	1000	0.91580	0.97286	0.03264	0.92049	0.91580
4	RF	F-test	500	0.91309	0.98571	0.03962	0.91457	0.91309
5	RF	LR-L2	500	0.90872	0.97857	0.04087	0.91358	0.90872
6	RF	Kruskal-Wallis	100	0.90831	0.98571	0.03754	0.90074	0.90831
7	RF	MI	1000	0.90644	0.98571	0.04287	0.91111	0.90644
8	RF	RF-based	1000	0.90582	0.99286	0.04368	0.90815	0.90582
9	RF	RF-based	500	0.90546	0.99286	0.04040	0.90815	0.90546
10	RF	MI	500	0.90494	0.98571	0.03869	0.91161	0.90494
11	RF	LR-L2	100	0.90451	0.95714	0.02570	0.90716	0.90451
	...							
24	RF	all-feat		0.85740	0.91857	0.03822	0.86914	0.85740
	N=5			**top 10**				
1	RF	LR-L2	500	0.91904	0.98571	0.03409	0.92543	0.91904
2	RF	LR-L2	1000	0.91330	0.98571	0.04382	0.92148	0.91330
3	RF	F-test	1000	0.91135	0.96571	0.03817	0.91506	0.91135
4	RF	Kruskal-Wallis	1000	0.90573	0.96571	0.03278	0.91605	0.90573
5	RF	F-test	500	0.90462	0.98571	0.04110	0.90667	0.90462
6	RF	Kruskal-Wallis	500	0.90342	0.98452	0.04161	0.90568	0.90342
7	SVM	all-feat		0.89874	0.95333	0.04683	0.91111	0.89874
8	RF	MI	1000	0.89783	0.95500	0.03978	0.90667	0.89783
9	RF	MI	500	0.89566	0.97857	0.04415	0.90420	0.89566
10	RF	RF-based	500	0.89443	0.97857	0.04581	0.90370	0.89443
11	RF	RF-based	100	0.88824	0.97143	0.04580	0.89383	0.88824
	...							
25	RF	all-feat		0.83819	0.90429	0.04519	0.85728	0.83819

TABLE 2. Top-10 best-performing configurations for each N-gram size ($N \in \{3, 4, 5\}$) ranked by average Accuracy (Acc). A configuration is defined by a classifier, a feature selection technique, and a feature subset size (50, 100, 500, or 1000 features). For each configuration, the averages of Acc, F1, and Recall values, computed over multiple runs, are reported, together her with maximum Acc achieved across runs and Acc standard deviation.

are highlighted in Table 2. Furthermore, the next four 3-gram highest-ranked configurations also achieved Acc values above 0.93, which is very close to the baseline. Overall, these findings demonstrate that, for certain classification models, with appropriate feature selection, we can use smaller feature sets and still have classification performance close to what can be obtained with the complete feature set.

The results indicate that RF is the most robust among the machine learning models, as every top-10 configurations that use feature-selection, across all values of N, are using RF. Random Forest algorithms benefit from the removal of N-grams with low informational power, since reducing the feature space to elements with high discriminative capacity increases the likelihood of selecting informative candidates for splitting, while simultaneously maintaining diversity.

We also found in our experiments that the best-performing configuration - 500 selected 3-gram features, F-test filtering and RF model, achieved perfect classification in one experimental run, accuracy, recall and F1 being equal to 1.0.

We also observe that, for each of the 3 sets of top 10 (with N having values 3, 4, 5), the results obtained by RF with feature selection are above the results obtained by RF by using all features. Moreover, it outperformed all other classification algorithms, except SVM with $N = 4$ on the full feature set. As for the number of selected features, in the top 10, there is no configuration with 50 selected features, and only four configurations in all 30 results in Table 2 using 100 selected features. We can conclude that, in these experiments, the best results are obtained for 500 and 1000 selected features. A study of the effect of feature set size on classification performance is presented in Section 5.

4.5. **Comparison with Previous Work.** In the literature, the best reported accuracies for authorship attribution often exceed 95% [14, 21]. However, variations in dataset composition, dimension, preprocessing procedures, evaluation protocols, experimental settings, etc. can have a substantial impact on performance. Therefore, such comparisons should be viewed just as a general point of reference.

Performance on datasets for authorship attribution in the Romanian language has improved over time. Using several machine learning models on the ROST dataset, [3] achieved a best accuracy of 80.94%, while subsequent studies [15, 16] obtained even stronger results, achieving an accuracy (there is referred as MAcc) above 95%, with several authors being classified perfectly. In the present study, the highest Acc reached 94.463%, and with highest F1 94.025% on the same configuration. This is a result that is comparable to,

although slightly lower than, the performance reported in the previously mentioned works. On same dataset, the BERT-based approach reported in [2] obtained 87.40% on the same 10-author corpus. The study presented in [18] uses a hybrid transformer architecture, and reported an F1-score of 87%, but it is applied to an extended version of the ROST dataset—which includes 19 authors, and so it addresses a harder problem.

One possible explanation for the better results obtained by traditional machine learning models is the limited size of the available corpus. Although modern neural network models can achieve excellent performance when trained on large-scale datasets, in case of a relatively small and imbalanced corpus as used in this study, comprising 400 documents from 10 authors, the neural models receive a relatively small amount of information for fine-tuning. This increases the risk of overfitting and limits the models' ability to identify reliable stylistic patterns.

5. The effect of feature set size on classification performance

We conducted additional experiments to evaluate performance across different feature set sizes (50, 100, 500, 1000, 1500, 2000, 2500, and 3000). Table 3 presents the average of Acc scores obtained for each classifier, N-gram order, and feature subset size. Across reported averages, RF attains the highest Acc, followed by kNN. Figure 3 is based on the average Acc scores computed for each combination of classifier, N-gram order, feature-subset size, and feature-selection method. The results reported there were obtained using kNN and RF, as these classifiers demonstrated the strongest overall performance. This analysis provides a clearer view of the impact of the different feature-selection methods and feature-set sizes on the two most competitive classification models.

As shown in Figure 3 (first column: A, C, and E) and in Table 3, the performance of kNN, measured in terms of Acc, generally declines as the number of selected features increases beyond 1000. In case of kNN, among the evaluated feature selection methods, variance and Chi2 filtering consistently yield the lowest Acc scores for the 4-gram and 5-gram representations. For the 3-gram representation, variance also performs poorly in most cases, while Chi2 achieves average results when more than 1000 features are used.

RF is the best-performing classification model when using feature-selection, as evidenced by the results reported in Table 3. Figure 3 shows that, in case of using RF, the Chi2 method yields superior results for larger feature subsets. The number of features beyond which Chi2 becomes the best-performing feature selection technique is 1000 features for N=3, 1500 features for N=4 and

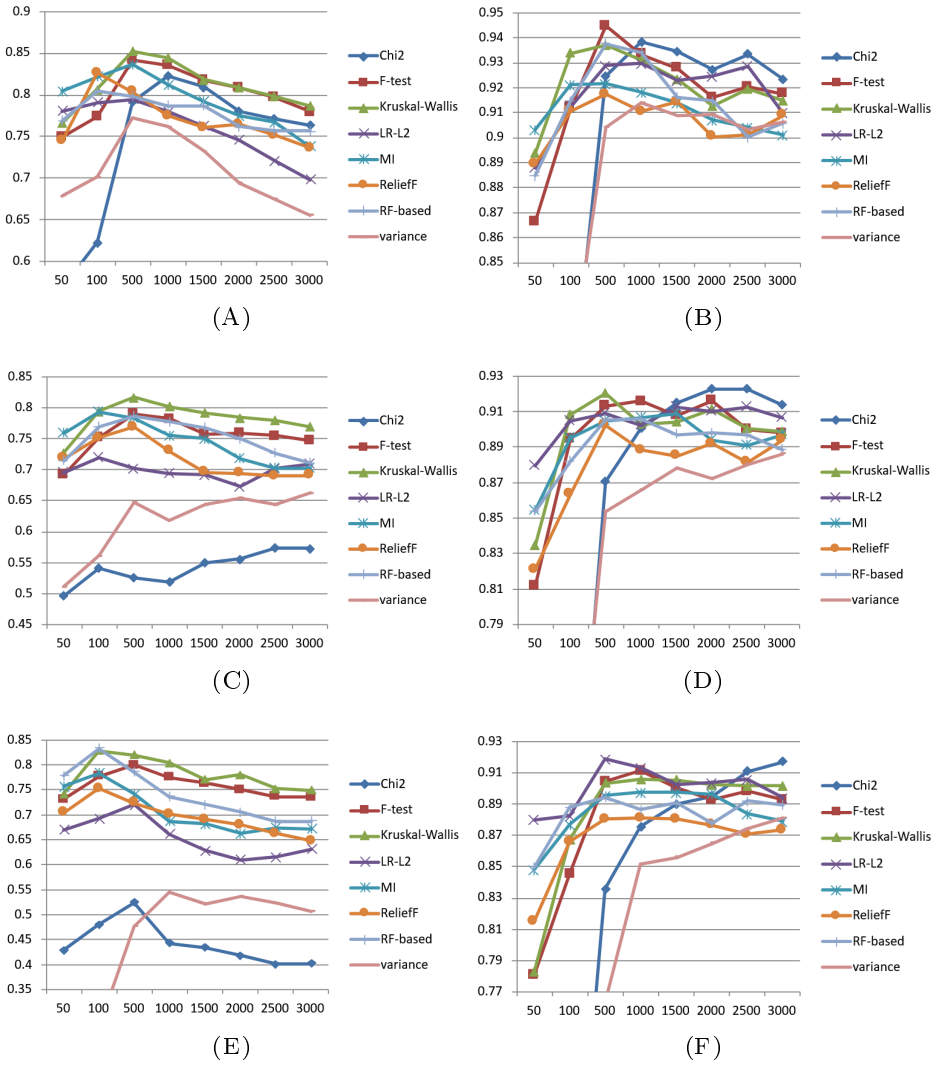


FIGURE 3. Average Acc scores (Y-axis) obtained for eight feature subset sizes (X-axis) using kNN and Random Forest (RF) classifiers. Results for kNN are presented in the first column (subfigures A, C, and E), while RF results are shown in the second column (subfigures B, D, and F). The experiments were conducted for N-gram sizes N=3 (subfigures A,B), N=4 (subfigures C,D) N=5 (subfigures E,F). The evaluated feature selection methods are listed on the right side of each subfigure. The average Acc scores were computed over multiple experimental runs for a N-gram size, a classifier, a feature selection method, and feature subset size.

	No. Features							
	50	**100**	**500**	**1000**	**1500**	**2000**	**2500**	**3000**
N=3								
kNN	0.73256	0.76862	0.81161	0.80229	0.78482	0.76735	0.75478	0.73936
LR	0.30768	0.40047	0.56066	0.63653	0.67683	0.70305	0.71966	0.73359
RF	0.85026	0.88820	0.92698	0.92615	0.92020	0.91399	0.91384	0.91096
SVM	0.13359	0.20829	0.46643	0.58912	0.66919	0.73157	0.76716	0.78688
N=4								
kNN	0.66439	0.71036	0.72749	0.70982	0.70592	0.69842	0.69661	0.69567
LR	0.23549	0.28815	0.43840	0.49935	0.53555	0.56526	0.58824	0.60528
RF	0.78225	0.82892	0.89735	0.89844	0.90107	0.90194	0.89811	0.89779
SVM	0.14215	0.17005	0.31602	0.37955	0.43154	0.48280	0.52302	0.56508
N=5								
kNN	0.62774	0.67794	0.69927	0.66920	0.65163	0.64311	0.63185	0.62913
LR	0.18450	0.21491	0.31802	0.35455	0.37593	0.40177	0.42469	0.44265
RF	0.71117	0.76848	0.87454	0.89049	0.89035	0.88881	0.89219	0.89116
SVM	0.12960	0.13989	0.22511	0.28159	0.30975	0.33254	0.34587	0.35897

TABLE 3. Average Acc scores obtained by kNN, LR, RF, and SVM for each of the feature subset sizes (50, 100, 500, 1000, 1500, 2000, 2500, and 3000) and character N-gram orders $N \in \{3, 4, 5\}$. Each cell reports the average Acc score for multiple runs for a classifier, N-gram order, and feature subset size.

2500 features for N=5. For lower number of selected features, Kruskal-Wallis, LR-L2, F-test are the best performers.

The optimal number of features is classifier-dependent. For kNN, a small, carefully chosen subset outperforms both smaller and larger subsets, optimum is at 500 features, at every N-gram order (Table 3). For RF, the optimal feature-set size varies with the N-gram order. In our experiments, the performance peak for 3-grams is reached with 500 or 1000 features, the two configurations producing nearly identical results. For 4-grams, the highest accuracy is obtained with 2000 features, although the results obtained with 1500 features are very similar. For 5-grams, the optimum is reached with 2500 features. Despite this variation, the best overall RF performance is obtained with 3-gram features. LR and SVM are monotone, in the sense that performance improves steadily with the size of the selected subset.

A short overview of the ten best-performing configurations identified in our experiments, ordered by Acc scores, is presented in Table 4. The first four positions are occupied by configurations using 500 or 1000 selected features,

while seven of the ten best results are obtained with a number of features less than or equal to 1000. Also, most of the positions in that table are occupied by 3-grams and RF classification model, with only one exception: row 5, where we have the results obtained by SVM by using the full set of 4-gram features. The table highlights the strong dominance of the RF classifier, which account for the vast majority of the top-ranked configurations. This observation is consistent with the overall results presented earlier, reinforcing the robustness of RF across different feature-selection settings.

No.	**Model**	**Feature Selection**	**No. Features**		**Acc avg**	**F1 avg**	**Recall avg**
1	RF	F-test	500	N=3	0.94463	0.94025	0.94463
2	RF	Chi2	1000	N=3	0.93834	0.93827	0.93834
3	RF	RF-based	500	N=3	0.93769	0.93728	0.93769
4	RF	Kruskal-Wallis	500	N=3	0.93704	0.93728	0.93704
5	SVM	all-feat		N=4	0.93443	0.93333	0.93443
6	RF	Chi2	1500	N=3	0.93436	0.93531	0.93436
7	RF	RF-based	1000	N=3	0.93428	0.92988	0.93428
8	RF	Kruskal-Wallis	100	N=3	0.93373	0.92840	0.93373
9	RF	Chi2	2500	N=3	0.93344	0.93531	0.93344
10	RF	F-test	1000	N=3	0.93331	0.93482	0.93331

TABLE 4. Top 10 configurations ranked by average Acc. The average values for Acc, F1, and Recall are calculated based on all runs performed for the specified configuration (classifier, feature selection method, number of features and character N-gram order).

6. CONCLUSIONS

This experimental study shows that feature selection can improve Romanian authorship attribution while reducing dimensionality. The results indicate that feature selection is particularly effective for the Random Forest (RF) classifier, with several selected-feature configurations outperforming the strongest model trained on the complete feature set, whereas other classifiers perform best when using the complete set of features.

The question of whether feature selection can achieve similar or better performance than the complete N-gram feature set is particularly relevant in the context of using N-gram sequences, because they generate a very large number of features, particularly for higher-order N-grams. In this study, the dimensionality reached approximately 269000 features for 5-grams, and even larger feature spaces can be expected for higher-order N-gram representations.

This study examines the results obtained by four classification algorithms, eight feature-selection techniques, character N-grams of sizes 3, 4, and 5, and multiple feature subset sizes. The experiments use repeated runs across five train-test splits, thereby enhancing the reliability of the results. Performance is compared against a baseline model that utilizes the complete feature set.

Among the baseline configurations, the best performance was achieved by the SVM classifier using 4-grams features, achieving an Acc of 0.93443. The best overall result we obtained was for a RF classifier combined with the F-test based feature filtering, using a subset of 500 features and a 3-gram representation, which achieved an Acc of 0.94463. In comparison, the best RF model trained on the complete feature set achieved an Acc of only 0.88327. Feature selection substantially improved the performance of RF, whereas it did not lead to performance gains in case of SVM model.

The present findings demonstrate that, by carefully choosing both the feature selection strategy and the classifier configuration, substantial dimensionality reduction can be achieved without sacrificing accuracy. Our experiments show that feature selection is not universally beneficial but is specifically advantageous for ensemble learners and distance-based models, while the other learners require broader feature space to recover their performance.

The absence of improvement for SVM contrasts with findings reported in previous studies on feature selection. We attribute this result to characteristics specific to our experimental setting. First, the full-feature SVM baseline already achieves a high performance level, leaving limited room for further gains. Second, SVMs are known to perform effectively in high-dimensional feature spaces, whereas the selected feature subsets are relatively small. Moreover, SVM scores improve as the number of selected features increases (Table 3). In contrast, RF, owing to its ensemble structure, inherent feature subsampling, and robustness to noisy variables, is better suited to benefit from aggressive feature reduction. Future work should investigate the interaction between feature selection and classification model performance (including SVM) across datasets of varying sizes and different classification scenarios.

Future work will focus on the generalization of our findings by conducting additional experiments on different datasets and for other languages too. Evaluating the proposed approach in multilingual settings would provide a better understanding of its robustness and its ability to capture language-independent patterns.

Another direction is to investigate whether feature selection techniques show different behaviors across various feature types. In particular, future studies could compare the effectiveness of feature selection when applied to lexical or

morpho-syntactic features. Such analysis could contribute to the development of more efficient and accurate models.

References

[1] Avram, S.-M. ROST (ROmanian Stories and other Texts). Kaggle dataset, 2022. https://www.kaggle.com/datasets/sandamariaavram/rost-romanian-stories-and-other-texts, last accessed September 2026.

[2] Avram, S.-M. BERT-based authorship attribution on the Romanian dataset called ROST. *Studia Universitatis Babeş-Bolyai Informatica 70*, 1–2 (2025), 43–60.

[3] Avram, S.-M., and Oltean, M. A comparison of several AI techniques for authorship attribution on Romanian texts. *Mathematics 10*, 23 (2022), 4589.

[4] Bolón-Canedo, V., Sánchez-Maroño, N., and Alonso-Betanzos, A. A review of feature selection methods on synthetic data. *Knowl. Inf. Syst. 34*, 3 (Mar. 2013), 483–519.

[5] Chandrashekar, G., and Sahin, F. A survey on feature selection methods. *Computers & Electrical Engineering 40*, 1 (2014), 16–28.

[6] Chaski, C. E. Who's at the keyboard? authorship attribution in digital evidence investigations. *Int. J. Digit. EVid. 4*, 1 (2005), 1–13.

[7] Fedotova, A., Romanov, A., Kurtukova, A., and Shelupanov, A. Authorship attribution of social media and literary Russian-language texts using machine learning methods and feature selection. *Future Internet 14*, 1 (2022), 4.

[8] Forman, G. An extensive empirical study of feature selection metrics for text classification. *Journal of Machine Learning Research 3* (2003), 1289–1305.

[9] Frantzeskou, G., MacDonell, S. G., and Stamatatos, E. Source code authorship analysis for supporting the cybercrime investigation process. In *Handbook of Research on Computational Forensics, Digital Crime, and Investigation: Methods and Solutions*, C.-T. Li, Ed. IGI Global, 2010, ch. 20, pp. 470–495.

[10] Georgieva-Trifonova, T., and Duraku, M. Research on n-grams feature selection methods for text classification. *IOP Conference Series: Materials Science and Engineering 1031*, 1 (2021), 012048.

[11] Guyon, I., and Elisseeff, A. An introduction to variable and feature selection. *Journal of Machine Learning Research 3* (2003), 1157–1182.

[12] He, X., Lashkari, A. H., Vombatkere, N., and Sharma, D. P. Authorship attribution methods, challenges, and future research directions: A comprehensive survey. *Information 15*, 3 (2024).

[13] Houvardas, J., and Stamatatos, E. N-gram feature selection for authorship identification. In *Artificial Intelligence: Methodology, Systems, Applications* (2006), pp. 77–86.

[14] Levchenko, M. Stylometry or embeddings? authorship attribution for russian and italian poetry. *Journal of Computational Literary Studies* (October 2026).

[15] Lupsa, D., Avram, S.-M., and Lupsa, R. Oldies but goldies: The potential of character N-grams for Romanian texts. *Studia Universitatis Babeş-Bolyai Informatica 70*, 1-2 (2025), 25–42.

[16] Lupsa, D., and Lupsa, R. Word and punctuation N-gram features in romanian authorship attribution. *Studia Universitatis Babeş-Bolyai Informatica 70*, 1-2 (2025), 61–74.

[17] Ma, S., and Huang, J. Penalized feature selection and classification in bioinformatics. *Briefings in Bioinformatics 9*, 5 (2008), 392–403.

[18] Nitu, M., and Dascalu, M. Authorship attribution in less-resourced languages: A hybrid transformer approach for Romanian. *Applied Sciences 14*, 7 (2024), 2700.

[19] Qin, J. Statistical feature selection and data-driven predictive model construction for intelligent decision-making. In *Proceedings of the 2026 International Conference on Artificial Intelligence and Fintech* (2026), IC-AIF '26, Association for Computing Machinery, p. 436–442.

[20] Ramnial, H., Panchoo, S., and Pudaruth, S. Authorship attribution using stylometry and machine learning techniques. In *Intelligent Systems Technologies and Applications* (2016), S. Berretti, S. M. Thampi, and P. R. Srivastava, Eds., Springer International Publishing, pp. 113–125.

[21] Sarwar, R., Yu, C., Tungare, N., Chitavisutthivong, K., Sriratanawilai, S., Xu, Y., Chow, D., Rakthanmanon, T., and Nutanong, S. An effective and scalable framework for authorship attribution query processing. *IEEE Access 6* (2018), 50030–50048.

[22] Stamatatos, E. A survey of modern authorship attribution methods. *Journal of the American Society for information Science and Technology 60*, 3 (2009), 538–556.

[23] Stamatatos, E. On the robustness of authorship attribution based on character n-gram features. *Journal of Law and Policy 21*, 2 (01 2013), 421–439.

[24] Urbanowicz, R. J., Meeker, M., La Cava, W., Olson, R. S., and Moore, J. H. Relief-based feature selection: Introduction and review. *Journal of Biomedical Informatics 85* (2018), 189–203.

[25] Yang, Y., and Pedersen, J. O. A comparative study on feature selection in text categorization. In *Proceedings of the Fourteenth International Conference on Machine Learning* (San Francisco, CA, USA, 1997), ICML '97, Morgan Kaufmann Publishers Inc., p. 412–420.

Computer Science Department, Babeş Bolyai University, Cluj-Napoca, Romania

Email address: dana.lupsa@ubbcluj.ro

Email address: radu.lupsa@ubbcluj.ro