Duzce University Journal of Science and Technology
Düzce Üniversitesi Bilim ve Teknoloji Dergisi
14 (1), 2026, 288-298.
https://doi.org/10.29130/dubited.1728460
e-ISSN: 2148-2446
https://dergipark.org.tr/en/pub/dubited

Review Article

Stylometric Profiling of Turkish Texts: Joint Estimation of Author, Region, Age and Genre

Vecdi Emre Levent[1*], Uğur Özbalkan[1]

[1] Fenerbahçe Üniversitesi Mühendislik ve Doğa Bilimleri Fakültesi, Bilgisayar Mühendisliği İngilizce, İstanbul, Türkiye. ror.org/00xf89h18

*Corresponding author:
emre.levent@fbu.edu.tr

Received: 30.06.2025
Revision: 28.10.2025
Accepted: 03.11.2025
Available Online: 21.01.2026

Abstract: Authorship identification seeks to determine the writer of a text by analyzing distinctive linguistic and stylistic features. These characteristics may vary across dimensions such as region, age, and genre. Identifying an author's stylistic fingerprint is essential in plagiarism detection, digital forensics, and computational linguistics. In this study, the authorship features of Turkish columnists were analyzed using Artificial Neural Networks (ANN), Support Vector Machines (SVM), and decision tree algorithms (J48 and Random Forest). Sixteen stylometric indicators were selected through the Zemberek natural language processing library and evaluated across six distinct datasets. The proposed system allows flexible parameter adjustment through a graphical interface and exports results in ARFF format for reproducibility. Experimental results demonstrated that Random Forest achieved the highest overall accuracy, particularly in regional and age-based datasets, with F-measures reaching up to 0.91. The accuracy rates were 73% for regional classification, 55% for genre classification, and 62.5% for age-based classification. The findings confirm that combining statistical learning with stylometric analysis provides a robust framework for Turkish authorship attribution, paving the way for future studies employing deep learning and transformer-based models.

Keywords: *Author recognition, Stylometry, Artificial neural network, Support vector machine, Decision trees, Authorship features*

I. INTRODUCTION

Validation and classification of textual documents are key subfields within natural language processing (NLP). When the task involves categorizing documents by type, the features employed may differ significantly from those used to identify an author. Although numerous studies have investigated authorship recognition in various languages, research on Turkish texts remains relatively limited. Artificial neural networks, machine learning algorithms, and statistical techniques are among the principal approaches used in this domain. The most critical aspect of any authorship recognition system is the selection of effective features capable of capturing an author's distinctive writing style (Amasyalı et al., 2006).

Research on authorship identification has explored a wide range of methods for detecting unique linguistic patterns. These include the use of synonym-based features (Clark & Hannon, 2007), linguistic profiling through extensive counts of grammatical and lexical indicators (van Halteren, 2004), and neural networks that learn stylistic cues from co-authors and metadata. Some of these methods have achieved accuracy rates exceeding 99% in authorship recognition tasks (van Halteren, 2004). The use of stylometric features has also yielded promising outcomes in author identification studies (Rocha et al., 2020). Furthermore, this concept has been extended to handwriting recognition, where author-specific characteristics have been shown to improve classification accuracy (Sas, 2006). Previous research has highlighted the distinctive role of synonym selection in distinguishing authors (Clark & Hannon, 2007). Such advances in authorship recognition technology pave the way for more sophisticated systems of intelligent text analysis and human–computer interaction.

The application of artificial neural networks (ANNs) to authorship attribution has increased substantially in recent years. In the field of information retrieval, researchers have examined various ANN architectures, including multilayer feed-forward networks (Selman & Husagic-Selman, 2011; Selman, 2012) and auto-associative networks (Umeda et al., 2002), to identify authors based on textual features. The effectiveness of these models in differentiating between authors has been demonstrated with high accuracy in several studies (Schifano et al., 2018; Selman, 2012). Numerous types of input features have been explored, such as word- and character-level statistics (Selman, 2012), stylometric markers (Stanczyk & Cyran, 2008) and directional index histograms for handwritten texts (Umeda et al., 2002). ANNs have been applied to multiple languages, including Turkish and English, and their use has expanded to related tasks such as signature verification and handwriting-based author identification (Remaida et al., 2020).

Cite as: Levent, V. E., & Özbalkan, U. (2026). Stylometric Profiling of Turkish Texts: Joint Estimation of Author, Region, Age and Genre. *Duzce University Journal of Science and Technology, 14*(1), 288-298. https://doi.org/10.29130/dubited.1728460

This is an open access paper distributed under the terms and conditions of the Creative Commons Attribution-NonCommercial 4.0 International License.

Catal et al. developed an n-gram-based model known as the NECL system (Çatal et al., 2003). Amasyalı and Diri proposed a grouping framework for author and document type recognition using twenty-two stylistic features (Amasyalı & Diri, 2006) and later applied 2-gram and 3-gram models (Amasyalı & Diri, 2006). Amasyalı and Yıldırım focused on classifying Turkish news articles (Amasyalı & Yıldırım, 2004), while Levent and Diri performed authorship recognition by extracting stylistic characteristics of Turkish columnists (Levent & Diri, 2014).

Deep learning has emerged as a transformative approach in stylometric analysis and authorship attribution, representing a significant advancement over traditional statistical methods. The application of sophisticated neural network architectures, including Convolutional Neural Networks, Recurrent Neural Networks, and transformer models, has opened new possibilities for understanding and identifying literary style and authorship in both historical and contemporary contexts Nishchal Bhargava et al. (Sharma & Kumar, 2024). These deep learning models demonstrate superior capability in capturing complex linguistic patterns and stylistic nuances that conventional approaches often overlook, leading to substantial improvements in accuracy and robustness.

Recent research has explored innovative methodologies such as combining deep neural networks across different stylometry feature categories, which has shown significant improvements in generalization ability and overall performance compared to single neural network approaches Pelin Canbay et al. (Canbay et al., 2020). Additionally, parallel stylometric document embeddings using deep learning-based language models have been successfully tested across multiple languages, with composite embedding methods consistently outperforming baseline approaches (Škorić et al., 2022). The field continues to evolve with the development of multilingual tools that enhance embedding layers for deep learning algorithms, proving valuable for various classification tasks and demonstrating the versatility of deep learning in stylometric applications Inez Okulska et al (Okulska et al., 2023). These advances have important implications for diverse fields ranging from literary studies and digital humanities to forensic linguistics and cybersecurity applications.

The distinguishing strength of the present study compared with (Amasyalı & Diri, 2006) lies in its flexibility. Owing to its adaptable design, the parameters of the Artificial Neural Network can be adjusted through a graphical user interface. Sixteen authorship features were extracted using the open-source Zemberek library (Akın & Akın, 2007). The proposed system is both user-friendly and extensible: it can be trained on different datasets and feature sets and can export results in ARFF (Attribute–Relation File Format) for use with Weka (Weka, n.d.). The implementation was carried out in C# within the .NET environment.

The remainder of this article is structured as follows: Section II presents the authorship features, Section III discusses the classification algorithms, Section IV details the author recognition application, Section V reports the experimental results, and Section VI concludes with findings and recommendations for future research.

II. SIGNATURE FEATURES OF AUTHORSHIP

We are so accustomed to the style and writings of the columnists we frequently follow that we can often guess who wrote an article when it appears elsewhere or remains unnamed. The aim of this study is to enable a computer to make this distinction automatically. Every columnist has a unique style, and if we can translate that style into a mathematical representation, we can perform author recognition. In the author recognition application, sixteen different features were selected as indicators of authorship.

Sixteen stylometric indicators were extracted per document to capture lexical richness, syntactic preference, and punctuation style. Sentence and token counts were obtained after Turkish tokenization and sentence splitting; from these, average word length (characters per token) and lexical diversity (type–token ratio computed as unique tokens divided by total tokens) were derived. Punctuation usage was characterized by the densities of periods, commas, semicolons, question marks, and exclamation marks (each divided by total tokens). A line-count feature was also included as a lightweight proxy for formatting and paragraphing style (normalized by tokens). Part-of-speech usage was represented by the ratios of nouns, verbs, adjectives, pronouns, adpositions, and conjunctions (each POS count divided by total tokens).

Because nouns, verbs, adjectives, pronouns, prepositions, and conjunctions are structural words, we used the open-source Zemberek [22] library developed for .NET. Figure 1

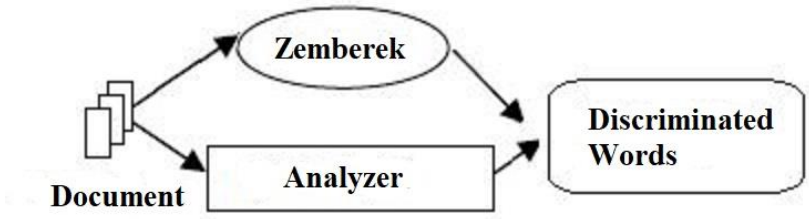


Figure 1. Feature Extraction Structure

Since the identified feature scales differ, normalization is essential. Varying feature scales can skew the system's performance. By scaling all values to the range [0, 1], each feature contributes equally to the system's discrimination.

III. CLASSIFICATION ALGORITHMS

Classification is one of the most commonly used methods in data mining. It involves assigning data instances to predefined classes within a data set. After learning from the training set, the model attempts to classify the test data correctly. A label is defined as the value that represents each data instance's class.

The workflow of classification algorithms begins by training on the given training data, then using the learned model to classify the test data.

A. Artificial Neural Networks

Artificial Neural Networks (ANNs), which are inspired by the brain's information-processing methods, create networks composed of layers of neurons. Numerical values are assigned to neurons and processed through transfer functions at each layer's outputs, with the final results obtained from the output layer. During training, the number of neurons, the number of layers, the choice of transfer functions, and the initial neuron values can be adjusted to observe their effects. The goal is to minimize the error between the target values and the network's outputs during each iteration.

A.1. Realization of Training

The identified authors and their corresponding documents are submitted to the system. The Artificial Neural Network then trains, minimizing the error until it reaches a specified threshold or the maximum number of iterations. During this process, the neuron weights for each layer are optimized. After providing the inputs and target values, the system monitors the error and epoch count and continues training until the stopping criteria are met.

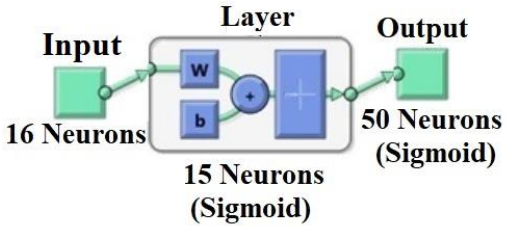


Figure 2. ANN Parameters

Figure 2 shows an example of an Artificial Neural Network model. It is a network containing 16 inputs, a hidden layer with 15 neurons, and 4 outputs corresponding to 4 different authors. All of these parameters can be selected by the user. The transfer function options include linear, sigmoid, proportional sigmoid, and Gaussian.

Transfer functions combine the inputs from connected neurons and produce numerical outputs according to the chosen mathematical function.

The learning rate is a key ANN parameter. It should be positive and close to zero. A smaller learning rate provides more stable convergence but slows down training, while a larger learning rate speeds up training but may prevent the network from reaching a low error. Finding a balance is important because training with a very small learning rate can become prohibitively slow, whereas a very large learning rate may never minimize the error (Elmas, 2003).

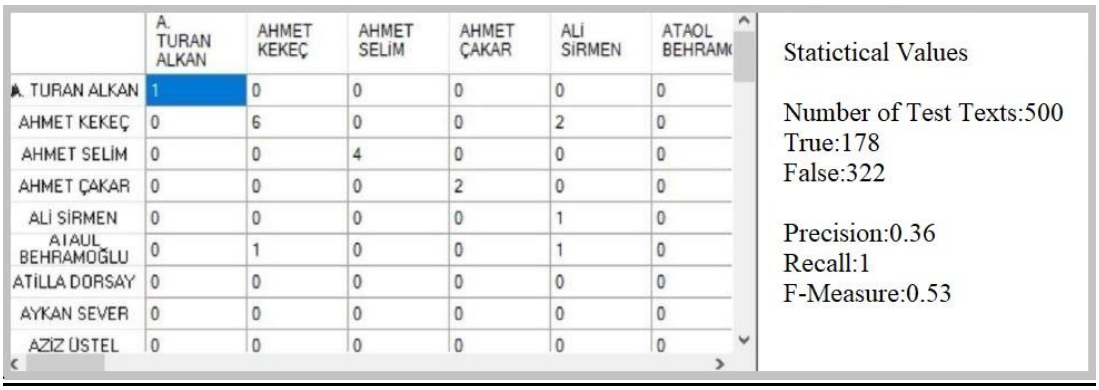
The momentum coefficient helps prevent the error from becoming stuck in local minima. Its value ranges from 0 to 1.

When momentum is zero, weight updates depend solely on the current gradient. As the momentum value increases toward 1, updates rely more on the previous weight changes and less on the current gradient. If the gradient direction remains consistent across two consecutive iterations, a higher momentum accelerates the weight update; otherwise, it

reduces the update magnitude. This mechanism smooths the learning trajectory and helps the network escape shallow local minima (Sağıroğlu et al., 2003).

A.2. Performing the Test

As a result of training, the network parameters are optimized and the model structure is established. To evaluate its performance, documents not used during training are classified and the outcomes are recorded. Our approach is as follows: if a neuron produces the highest output value, the author corresponding to that neuron is chosen as the author of the input document.



	A. TURAN ALKAN	AHMET KEKEÇ	AHMET SELİM	AHMET ÇAKAR	ALİ SİRMEN	ATAOL BEHRAM
A. TURAN ALKAN	1	0	0	0	0	0
AHMET KEKEÇ	0	6	0	0	2	0
AHMET SELİM	0	0	4	0	0	0
AHMET ÇAKAR	0	0	0	2	0	0
ALİ SİRMEN	0	0	0	0	1	0
ATAOL BEHRAMOĞLU	0	1	0	0	1	0
ATİLLA DORSAY	0	0	0	0	0	0
AYKAN SEVER	0	0	0	0	0	0
AZİZ ÜSTEL	0	0	0	0	0	0

Statictical Values

Number of Test Texts:500
True:178
False:322

Precision:0.36
Recall:1
F-Measure:0.53

Figure 3. Confusion Matrix for a Representative Subset of Six Authors

The full confusion matrix is computed and displayed within the graphical user interface for interactive inspection. However, because DS1 contains 50 authors, a full-size matrix would be visually dense and unreadable at publication scale. Therefore, Figure 3 shows a representative screenshot from the application's UI, illustrating the class-wise error patterns, precision, recall, and F-measure results.

B. Support Vector Machine

Support Vector Machines (SVMs) are supervised learning methods used for both regression and classification tasks. The concept of the SVM was first introduced by Vladimir Vapnik and later refined by Corinna Cortes and Vapnik. Although early SVM research focused on classification, it was found to yield strong performance in regression applications as well.

Given a training set D = {(x1, y1), (x2, y2), . . . , (xi, yi)}, where each xi is an N -dimensional input vector and yi is the corresponding output, SVM seeks a function of the form f (x) = w · x + b, where w is the normal vector, x is the input vector, and b is a bias term.

In support vector regression (SVR), the goal is to find a function f (x) that predicts output values within a maximum allowable error. Simultaneously, SVR minimizes the distance between two parallel hyperplanes that enclose the training data, ensuring that most data points lie within this margin.

C. Decision Trees

Decision trees are a highly versatile and well-regarded predictive model that is employed in a variety of applications, including classification, regression and other tasks in machine learning and data mining (Kliegr et al., 2020; Rokach, 2016). The primary reason for using decision trees is their simple, interpretable structure. Another advantage is that training time is faster compared to many other algorithms. Decision trees are built using training data and then evaluated on test data. These evaluations can inform future studies and support more accurate predictions (Birant, 2011; García-Laencina et al., 2010).

C.1. J48

The J48 algorithm, a Java implementation of the C4.5 decision tree classifier, is widely used in data mining for classification tasks (Arora & Suman, 2012; Singaravelan et al., 2015). It has been successfully applied in diverse domains such as cancer recurrence prediction (Bhargava et al., 2017), student performance analysis (Tun & Htay, 2020), and examination success rate forecasting (Clarin, 2020). Comparative studies have shown that J48 performs competitively against other algorithms, including Sequential Minimal Optimization (SMO) and Multilayer Perceptron (MLP), across various datasets (Arora & Suman, 2012; Singaravelan et al., 2015). Efforts to enhance the performance of J48 have included its integration with Kendall's Rank Correlation Coefficient (Saravanan & Gayathri, 2018) and the application of data preprocessing techniques such as discretization (Chandrasekar et al., 2017). Overall, J48 remains a popular and effective algorithm for classification problems across multiple research and application areas.

C.2. Random Forest

Random Forest (RF) is a powerful ensemble learning algorithm commonly used for both classification and regression tasks (Breiman, 2001). It combines multiple decision trees, each trained on bootstrap samples and random subsets of features, to generate predictions through majority voting or averaging (Svetnik et al., 2003). RF offers several advantages, including non-parametric modeling, effective handling of high-dimensional data, and robustness to noise and outliers (Horning, 2010).

IV. EXPERIMENTEL RESULTS

In the Author Recognition study, six datasets (DS1–DS6) were constructed, as summarized in Table 1. DS1 was composed of 50 articles by 16 authors across different categories; 40 articles were allocated to training and 10 to testing (overall). DS2 consisted of 400 articles produced by 10 male and 10 female authors; 320 were used for training and 80 for testing. DS3 was formed within a single category, with 50 articles per author for 16 authors; for each author, 40 were assigned to training and 10 to testing (total 800). DS4 was organized by region, with 50 documents per region and a 40/10 train–test split per region. In DS5, documents were grouped by news type (Magazine, Politics, Sports, Life). In DS6, documents were grouped into age ranges (35–55, 55–65, 65–75, 75–90).

Table 1. Summary of Datasets Used for Training and Testing

DS	Task / Label	Class Definition	#Classes	Docs per class	Total docs
DS1	Author Recognition	Author	16	50	800
DS2	Gender classification	Male / Female (10+10 authors)	2	200	400
DS3	Author identification (same category)	Author	16	50	800
DS4	Region classification	Region	10	50	500
DS5	Genre classification	4 genres	4	100	400
DS6	Age-group classification	4 bins (35–55, 55–65, 65–75, 75–90)	4	100	400

A. Artificial Neural Network Structure Used in Training

The Artificial Neural Network used has a structure with 16 inputs and three hidden layers of 50 neurons each, all employing a sigmoid transfer function corresponding to the 16 authorship features. Figure 4 illustrates the ANN structure designed for 16 classes of data.

Since the learning rate and momentum are parameters of the artificial neural network, both were set to 0.1. The minimum error threshold and the maximum number of iterations can be adjusted by the user. By default, the maximum number of iterations is 30,000, and the minimum error threshold is 0.1. Because we use Weka for further processing, the data can be saved in ARFF (Attribute-Relation File Format). The trained network is also saved as an .xml file for use during the testing phase.

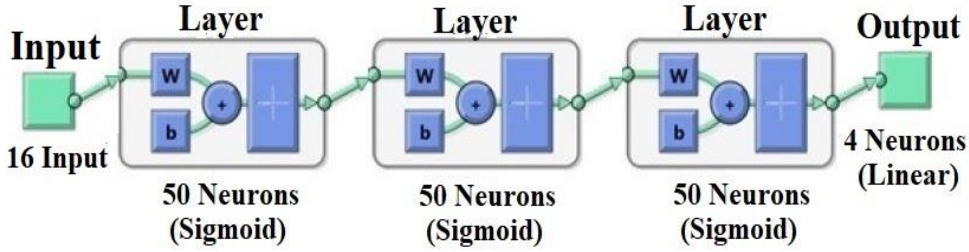


Figure 4. ANN Structure and Parameters

B. Test Results

For our first data set, DS1, experiments were conducted with different numbers of classes; the resulting F-measure and error rate are shown in Table 2.

Table 2. Author Identification: Performance by Author Count

#Author	F-Measurement	Error Rate	#Loop	Duration (min)
4	1	0,09	2792	3,28
8	0,78	0,09	11119	27,46
12	0,95	0,09	11619	44,37
16	0,73	0,1	13768	71,58

When examining Table 2, we observe that performance declines as the number of classes increases. The reason why the 12-author case outperforms the 8-author case is that the authorship characteristics of those four additional authors are more distinctive.

For the DS2 dataset, the results are shown in Table 3.

Table 3. Gender Classification

F-Measurement	Error Rate	#loop	Duration (min)
0,75	0,16	35000	204.44

From the results in Table 3 for the 10 male and 10 female authors, the detection accuracy for male authors was higher. This suggests that male authors share more similar writing characteristics, whereas female authors exhibit greater variation.

The results for the DS3 dataset are presented in Table 4.

Table 4. Author Identification Single Category: Performance by Author Count

#Author	F-Measurement	Error Rate	#loop	Duration (min)
4	0,98	0,1	5334	10.41
8	0,98	0,1	12876	51.0
12	0,98	0,1	14802	86.29
16	0,83	0,18	50000	262.2

The performance on the DS3 dataset—composed of authors writing within the same category—exceeded that of DS1. Because these authors' styles were less similar to each other, the classes could be separated more effectively. To compare our results with those from the Artificial Neural Network, we exported the feature files as ARFF and ran them through the SVM implementation in Weka. Table 5 shows the results obtained on DS1 and their comparison with the Artificial Neural Network.

Table 5. ANN vs SVM: F-measure by Class Count

#Author	Author Recognizer (ANN F-Measurement)	Weka - SVM (F-Measurement)
4	1	0,95
8	0,78	0,75
12	0,95	0,84
16	0,73	0,75

The comparative results for our second data set, DS2 (gender-dependent), are shown in Table 6.

Table 6. ANN vs SVM: F-measure

Author Recognizer (ANN F-Measurement)	Weka - SVM (F-Measurement)
0,75	0,75

The results from our gender-based data set match those shown in Table 6. The comparative results obtained for DS3, our third data set consisting of authors from the same category, are shown in Table 7.

Table 7. DS3 ANN-SVM Comparison

#Author	Author Recognizer (ANN F-Measurement)	Weka - SVM (F-Measurement)
4	0,98	1
8	0,98	0,92
12	0,98	0,84
16	0,83	0,88

In our DS3 dataset, as in DS1, the SVM achieved better results when classifying 16 authors.

Training the system with an ANN during experimental studies takes a long time, but once the network is trained, the testing process is very fast. Using the same network model for classifications ranging from 2 to 16 classes meant that employing a smaller number of classes was unnecessary.

Beyond these three data sets, we also ran experiments on data sets organized by region, news type, and age. In the DS4 dataset, authors are divided by region, with each region providing 50 documents. Of these, 40 documents per region were used for training and 10 for testing.

Table 8 presents the DS4 results for all algorithms (ANN, SVM, J48, and Random Forest) in a single consolidated view, reporting F-measure, Precision, and Recall. Random Forest achieves the highest F-measure, while SVM yields the lowest. Presenting all metrics together improves readability and enables a consistent side-by-side comparison across methods.

Table 8. Region Classification

Algorithm	F-measure	Precision	Recall
ANN	0.79	0.65	1
SVM	0.59	0.72	0.61
J48	0.72	0.76	0.73
Random Forest	0.91	0.92	0.91

In our DS5 data set, authors are separated by news type—Magazine, Politics, Sports, and Life—and documents are assigned to each genre. Table 9 shows the results of running this data set with the ANN, SVM, J48, and Random Forest algorithms.

Table 9. Genre Classification

Algorithm	F-Measurement	Precision	Recall
ANN	0.90	0.83	1
SVM	0.82	0.82	0.82
J48	0.50	0.54	0.55
Random Forest	0.54	0.57	0.57

As shown in the tables, different algorithms perform better on different data sets. These performance differences arise from each algorithm's logic and its compatibility with the respective data set.

For the DS6 dataset—grouping authors into age ranges of 35–55, 55–65, 65–75, and 75–90—the results obtained using these algorithms are presented in Table 10.

Table 10. Age Group Classification

Algorithm	F-Measurement	Precision	Recall
ANN	0.57	0.40	1
SVM	0.44	0.45	0.45
J48	0.62	0.63	0.62
Random Forest	0.90	0.91	0.90

The Random Forest algorithm is preferred for its simplicity, flexibility, and effectiveness in both classification and regression, and in our experiments it yielded the best results on the DS4 and DS6 datasets. Different outcomes were observed when authors were grouped according to various criteria. Authors who live in the same region and share similar ages tend to write the same type of news; moreover, they exhibit similar stylistic features and characteristics.

Findings related to gender and age were interpreted as associations rather than causal effects and should be considered within the constraints of this corpus. Demographic labels originate from editorial metadata and may include uncertainty; class imbalance and topic overlap can affect measured performance. Generalizability is limited to this dataset and time period.

The overall pattern is visualized in Figure 5, where the best F-measure per dataset is summarized.

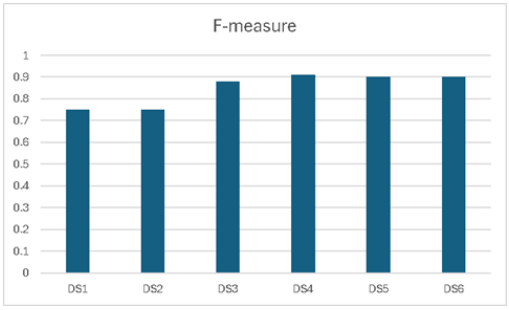


Figure 5. F-Measure with Best Algorithms

As summarized in Table 11, model rankings varied across datasets. Random Forest yielded the highest F-measure on DS4 (0.91) and DS6 (0.90), ANN led DS5 (0.90), and SVM led DS1 (0.75) and DS3 (0.88), while DS2 showed a tie (ANN = SVM, 0.75). Where reported, Precision and Recall are included; dashes indicate metrics not reported for those datasets.

Table 11. Best Model per Dataset

Dataset	Task (Label)	Best Model	Key Configuration	F-measure
DS1	Author (16 authors)	SVM	kernel linear, C 1.0, tol 1e-12, eps 0.001, no scaling	0.75
DS2	Gender (M/F, 10+10 authors)	ANN = SVM	ANN 16-50-50-50-Out, lr 0.1, mom 0.1 | SVM kernel linear, C 1.0, tol 1e-12, eps 0.001	0.75
DS3	Author (same category, 16 authors)	SVM	kernel linear, C 1.0, tol 1e-12, eps 0.001, no scaling	0.88
DS4	Region (10 classes)	Random Forest	trees 100, features √M, max depth none, bag 100%, seed 1	0.91
DS5	Genre (4 classes)	ANN	hidden 16-50-50-50-Out, lr 0.1, mom 0.1	0.9
DS6	Age (4 bins)	Random Forest	trees 100, features √M, max depth none, bag 100%, seed 1	0.9

Across datasets, the observed performance gaps can be explained by class balance, class separability, feature diversity, and model capacity. Region- and age-grouped tasks exhibit stronger stylistic cues, favoring Random Forest; genre structure benefits ANN, while SVM trails under non-linear boundaries. These factors align with the patterns summarized in Table 11, clarifying why different algorithms lead on different datasets.

Experiments were evaluated with stratified 5-fold cross-validation. A small grid search over standard hyperparameters was performed on the training split in each fold, and the best setting was used for testing. Reported scores are mean macro values across folds. Pairwise differences between top methods were checked with a McNemar test on pooled fold predictions at $p < 0.05$.

V. CONCLUSION

In the realm of authorship attribution, research focused on the Turkish language has remained limited compared to other languages such as English or Spanish. This study bridges that gap by presenting a comprehensive experimental framework that integrates statistical and machine learning techniques for stylometric profiling of Turkish texts. Six distinct datasets were designed to examine how authorship features vary across demographic and thematic dimensions—such as author identity, gender, writing category, geographical region, news type, and age group.

The experimental findings highlight the robustness of ensemble and neural models in capturing the stylistic nuances of Turkish writing. Across the six datasets, Random Forest most frequently ranked as the top performer—reaching F-measure = 0.91 in region- and age-grouped settings—while SVM generally trailed ensemble and neural approaches; ANN remained competitive in genre-focused tasks. These differences indicate that no single model universally dominates; rather, algorithm suitability depends on dataset structure, feature diversity, and class separability.

These results validate feature-based approaches for Turkish authorship analysis and establish a reproducible baseline for future studies. The system's modular design—allowing parameter adjustments and dataset customization—facilitates rapid experimentation and scalability. Furthermore, ARFF export supports interoperability with established tools such as Weka, enabling further comparative analyses.

Future research directions include extending the framework with deep learning and transformer-based architectures (e.g., BERTurk/TurkishBERT), n-gram embeddings, hybrid CNN–LSTM models, and transfer learning. Incorporating statistical significance tests and cross-validation will strengthen reliability, while expanding the datasets to social-media and literary texts will enhance generalizability.

In summary, combining traditional stylometric analysis with modern machine learning yields strong, interpretable, and scalable results, providing a practical default (ensemble methods) and a clear path for multilingual and cross-domain authorship research.

DECLARATIONS

Acknowledgements: The authors do not wish to acknowledge any individual or institution.

Author Contributions: Both authors contributed equally to all stages of the study, including conceptualization, methodology, data curation, investigation, analysis, writing—original draft preparation, writing—review and editing, and supervision. All authors have read and approved the final version of the manuscript.

Conflict of Interest: The authors declare no conflict of interest.

Supporting Institutions: This research received no external funding.

Ethical Approval: This study does not involve human or animal participants. All procedures followed scientific and ethical principles, and all referenced studies are appropriately cited.

Plagiarism Statement: This article has been evaluated for plagiarism and no instances of plagiarism were detected.

Data Availability: Data sharing is not applicable to this study.

Use of AI Tools: The authors declare that no Artificial Intelligence (AI) tools were used in the creation of this article.

REFERENCES

Akın, A. A., & Akın, M. D. (2007). Zemberek: An open source NLP framework for Turkic languages. *Structure, 10*, 1-5. https://www.academia.edu/download/34521696/zemberek_makale.pdf

Amasyalı, M. F., & Yıldırım, T. (2004). Otomatik haber metinleri sınıflandırma. In *Proceedings of the IEEE 12th Signal Processing and Communications Applications Conference (SIU 2004)* (pp. 224–226). *IEEE.*

Amasyalı, M. F., & Diri, B. (2006). Automatic Turkish text categorization in terms of author, genre and gender. In *International Conference on Application of Natural Language to Information Systems* (pp. 221-226). Springer. https://doi.org/10.1007/11765448_22

Amasyalı, M. F., Diri, B., & Türkoğlu, F. (2006). Farklı özellik vektörleri ile Türkçe dokümanların yazarlarının belirlenmesi. In *15th Turkish Symposium on Artificial Intelligence and Neural Networks* (pp. 1–4). Muğla, Türkiye.

Arora, R., & Suman, S. (2012). Comparative analysis of classification algorithms on different datasets using WEKA. *International Journal of Computer Applications*, *54*(13), 21-25.

Birant, D. (2011). Comparison of decision tree algorithms for predicting potential air pollutant emissions with data mining models. *Journal of Environmental Informatics*, *17*(1), 46–53. https://doi.org/10.3808/jei.201100186

Bhargava, N., Sharma, S., Purohit, R., & Rathore, P. S. (2017). Prediction of recurrence cancer using J48 algorithm. In *Proceedings of the 2017 2nd International Conference on Communication and Electronics Systems (ICCES 2017)* (pp. 386–390). IEEE. https://doi.org/10.1109/CESYS.2017.8321306

Breiman, L. (2001). Random forests. *Machine Learning*, *45*(1), 5-32. https://doi.org/10.1023/A:1010933404324

Canbay, P., Sezer, E. A., & Sever, H. (2020). Deep combination of stylometry features for authorship analysis. *International Journal of Information Security Science*, *9*(3), 154-163.

Chandrasekar, P., Qian, K., Shahriar, H., & Bhattacharya, P. (2017). Improving the prediction accuracy of decision tree mining with data preprocessing. In *2017 IEEE 41st Annual Computer Software and Applications Conference (COMPSAC)* (Vol. 2, pp. 481-484). IEEE. https://doi.org/10.1109/COMPSAC.2017.146

Clarin, J. A. (2020). J48-based algorithm in predicting the success rate in the board examination. *International Journal*, *9*(1.3), 122-127. https://doi.org/10.30534/ijatcse/2020/1791.32020

Clark, J. H., & Hannon, C. J. (2007). A classifier system for author recognition using synonym-based features. In *Mexican International Conference on Artificial Intelligence* (pp. 839-849). Springer. https://doi.org/10.1007/978-3-540-76631-5_80

Çatal, Ç., Erbakırcı, K., & Erenler, Y. (2003). Computer-based authorship attribution for Turkish documents. In *Turkish Symposium on Artificial Intelligence and Neural Networks* (pp. 539-541).

Elmas, Ç. (2003). *Yapay sinir ağları.* Seçkin Yayınevi.

García-Laencina, P. J., Sancho-Gómez, J. L., & Figueiras-Vidal, A. R. (2010). Pattern classification with missing data: A review. *Neural Computing and Applications*, *19*(2), 263-282. https://doi.org/10.1007/s00521-009-0295-6

Horning, N. (2010). Random Forests: An algorithm for image classification and generation of continuous fields data sets. In *Proceedings of the International Conference on Geoinformatics for Spatial Infrastructure Development in Earth and Allied Sciences* (pp. 1–6). Osaka, Japan.

Kliegr, T., Bahník, Š., & Fürnkranz, J. (2020). Advances in machine learning for the behavioral sciences. *American Behavioral Scientist, 64*(2), 145-175. https://doi.org/10.1177/0002764219859639

Levent, V. E., & Diri, B. (2014). Türkçe dokümanlarda yapay sinir ağları ile yazar tanıma. In *Akademik Bilişim 2014 Bildiriler Kitabı*, (pp. 735-741).

Okulska, I., Stetsenko, D., Kołos, A., Karlińska, A., Głąbińska, K., & Nowakowski, A. (2023). Stylometrix: An open-source multilingual tool for representing stylometric vectors*. arXiv.* https://arxiv.org/abs/2309.12810

Remaida, A., Moumen, A., El Idrissi, Y. E. B., & Sabri, Z. (2020). Handwriting recognition with artificial neural networks a decade literature review. *Proceedings of the 3rd International Conference on Networking, Information Systems & Security* (Article 65, pp. 1–5). https://doi.org/10.1145/3386723.3387884

Rocha, M. A. D., Nóbrega, G. Â. S. D., de Medeiros Valentim, R. A., & Alves, L. P. C. (2020). A text as unique as fingerprint: AVASUS text analysis and authorship recognition. In *Euro-American Conference on Telematics and Information Systems* (pp. 1-8). ACM.

Rokach, L. (2016). Decision forest: Twenty years of research. *Information Fusion*, *27*, 111-125. https://doi.org/10.1016/j.inffus.2015.06.005

Sağiroğlu, Ş., Erler, M., & Beşdok, E. (2003). *Mühendislikte yapay zeka uygulamaları-I: Yapay sinir ağları*. Ufuk Kitabevi.

Saravanan, N., & Gayathri, V. (2018). Performance and classification evaluation of J48 algorithm and Kendall's based J48 algorithm (KNJ48). *International Journal of Computer Trends and Technology*, *59*(1), 73-80. https://doi.org/10.14445/22312803/IJCTT-V59P112

Sas, J. (2006). Handwriting recognition accuracy improvement by author identification. In *International Conference on Artificial Intelligence and Soft Computing* (pp. 682-691). Springer.

Schifano, S. F., Sgarbanti, T., & Tomassetti, L. (2018). Authorship recognition and disambiguation of scientific papers using a neural networks approach*. Proceedings of Science,* 327, 007. https://doi.org/10.22323/1.327.0007.

Selman, S. (2012). Distinction of the authors of texts using multilayered feedforward neural networks. *Southeast Europe Journal of Soft Computing*, *1*(1), 128-138.

Selman, S., & Husagic-Selman, A. (2011). Multilayered feedforward neural networks as a tool for distinction of the authors of texts. In *International Symposium on Information, Communication and Automation Technologies* (pp. 1-6). IEEE.

Sharma, N., & Kumar, A. (2024). Deep learning for stylometry and authorship attribution: A review of literature. *International Journal for Research in Applied Science and Engineering Technology*, *12*(9), 212-215. https://doi.org/10.22214/ijraset.2024.64168

Singaravelan, S., Murugan, D., & Mayakrishnan, R. (2015). Analysis of classification algorithms J48 and Smo on different datasets. *World Engineering & Applied Sciences Journal*, *6*(2), 119-123.

Škorić, M., Stanković, R., Ikonić Nešić, M., Byszuk, J., & Eder, M. (2022). Parallel stylometric document embeddings with deep learning based language models in literary authorship attribution. *Mathematics*, *10*(5), Article 838. https://doi.org/10.3390/math10050838

Stanczyk, U., & Cyran, K. A. (2008). Application of artificial neural networks to stylometric analysis. In *Proceedings of the 8th Conference on Systems Theory and Scientific Computation* (pp. 25-30). https://dl.acm.org/doi/abs/10.5555/1503773.1503779

Svetnik, V., Liaw, A., Tong, C., Culberson, J. C., Sheridan, R. P., & Feuston, B. P. (2003). Random forest: A classification and regression tool for compound classification and QSAR modeling. *Journal of Chemical Information and Computer Sciences*, *43*(6), 1947-1958. https://doi.org/10.1021/ci034160g

Tun, M. T., & Htay, Y. Y. (2020). Predict Students' Performance by Using J48 Algorithm. *International Journal of Scientific Research in Science, Engineering and Technology*, *7*(3), 578-582. https://doi.org/10.32628/IJSRSET2073124

Umeda, M., Miyoshi, T., & Misaki, K. (2002). Writer identification and verification using autoassociative neural networks. *IEEJ Transactions on Electronics, Information and Systems*, *122*(11), 1869-1875. https://doi.org/10.1541/ieejeiss1987.122.11_1869

van Halteren, H. (2004). Linguistic profiling for authorship recognition and verification. In *Proceedings of the 42nd Annual Meeting of the Association for Computational Linguistics (ACL-04)* (pp. 199-206). Association for Computational Linguistics. https://doi.org/10.3115/1218955.1218981

Weka. (n.d.). *Weka 3: Data mining software in Java.* University of Waikato. Retrieved May 2025, from http://www.cs.waikato.ac.nz/ml/weka/