LITERARY FINGERPRINTS: DETECTION OF BOOK GENRE AND AUTHOR STYLE USING NATURAL LANGUAGE PROCESSING AND MACHINE LEARNING

DOĞUKAN ÖZYURT

Thesis for the Master's Program in Electrical and Electronics Engineering

Graduate School
Izmir University of Economics
Izmir
2025

LITERARY FINGERPRINTS: DETECTION OF BOOK GENRE AND AUTHOR STYLE USING NATURAL LANGUAGE PROCESSING AND MACHINE LEARNING

DOĞUKAN ÖZYURT

THESIS ADVISOR: ASST. PROF. DR. AYÇA KUMLUCA TOPALLI

Master's Exam Jury Members
Assoc. Prof. Dr. Nalan Özkurt
Assoc. Prof. Dr. Mehmet Türkan
Asst. Prof. Dr. Ayça Kumluca Topallı

A Master's Thesis
Submitted to
the Graduate School of Izmir University of Economics
the Department of Electrical and Electronics Engineering

Izmir
2025

Approval of the Graduate School

Prof. Dr. Mehmet Efe BİRESSELİOĞLU

I certify that this thesis satisfies all the requirements as a thesis for a Master's degree.

Prof. Dr. Aydın AKAN

This is to certify that we have read this thesis and that in our opinion it is fully adequate, in scope and quality, as a thesis for a Master's degree.

Asst. Prof. Dr. Ayça KUMLUCA TOPALLI

Master's Exam Jury Members

Assoc. Prof. Dr. Nalan ÖZKURT
Yaşar University

Assoc. Prof. Dr. Mehmet TÜRKAN
Izmir University of Economics

Asst. Prof. Dr. Ayça KUMLUCA TOPALLI
Izmir University of Economics

ETHICAL DECLARATION

I hereby declare that I am the sole author of this thesis and that I have conducted my work in accordance with academic rules and ethical behaviour at every stage from the planning of the thesis to its defence. I confirm that I have cited all ideas, information and findings that are not specific to my study, as required by the code of ethical behaviour, and that all statements not cited are my own.

Name, Surname: Doğukan ÖZYURT

Date: 19.06.2025

Signature:

ABSTRACT

LITERARY FINGERPRINTS: DETECTION OF BOOK GENRE AND AUTHOR STYLE USING NATURAL LANGUAGE PROCESSING AND MACHINE LEARNING

Özyurt, Doğukan

Master's Program in Electrical and Electronics Engineering

Advisor: Asst. Prof. Dr. Ayça Kumluca Topallı

June, 2025

Authorship attribution, genre classification, and recommendation systems based on literary content have become increasingly important in natural language processing (NLP). This thesis presents a system capable of both classification and content-based recommendation by representing book texts in a vector space. Comprehensive datasets of Turkish and English novels were compiled. All texts were lowercased and preprocessed to remove punctuation and special characters, then transformed into fixed-length vectors using the Doc2Vec algorithm. To ensure generalization, a custom train/test split was applied, guaranteeing each author was represented in the test set. These document vectors were used for authorship and genre classification with classifiers such as Logistic Regression, LinearSVC, Random Forest, Gaussian Naive Bayes, and K-Nearest Neighbors. This study also adopts a centroid-based similarity approach that extends traditional cosine similarity principles. For each author and genre, a representative vector was computed from the training data, and test books

were classified based on their angular proximity to these centroids. This approach improved conceptual sensitivity and yielded high accuracy. In the second phase, a content-based recommendation system was developed that, unlike collaborative filtering, relies solely on textual similarity. It recommends books with similar narrative styles to those already enjoyed by the reader, helping raise the visibility of lesser-known authors. This research shows that document embedding-based methods can be effectively applied in digital humanities, authorship attribution, and recommendation systems.

Keywords: Doc2Vec, Machine Learning, Natural Language Processing, Genre Classification, Text Classification, Authorship Attribution

ÖZET

EDEBİ PARMAK İZLERİ: DOĞAL DİL İŞLEME VE MAKİNE ÖĞRENMESİ YÖNTEMLERİYLE KİTAP TÜRÜ VE YAZAR ANLATIM TARZININ BELİRLENMESİ

Özyurt, Doğukan

Elektrik Elektronik Mühendisliği Yüksek Lisans Programı

Tez Danışmanı: Dr. Öğr. Üyesi Ayça Kumluca Topallı

Haziran, 2025

Edebi eserlerin içeriklerine dayalı olarak yazar tahmini yapmak, tür sınıflandırması gerçekleştirmek ve öneri sistemi geliştirmek, doğal dil işleme alanında giderek önem kazanan bir araştırma konusudur. Bu tez çalışmasında, kitap metinlerinin vektör uzayında temsil edilmesiyle hem sınıflandırma hem de içerik tabanlı öneri sunabilen bir sistem geliştirilmiştir. Çalışma kapsamında Türkçe ve İngilizce romanlardan oluşan kapsamlı veri kümeleri oluşturulmuştur; metinler küçük harfe dönüştürülmüş, noktalama işaretleri ve özel karakterlerden temizlenerek ön işleme tabi tutulmuştur. Temizlenen metinler, Doc2Vec algoritması ile sabit boyutlu vektör temsillerine dönüştürülmüştür. Modelin genelleme başarımını değerlendirmek amacıyla, her yazarın en az bir kitabının test kümesine dahil edildiği özel bir eğitim/test ayrımı uygulanmıştır. Elde edilen belge vektörleri, Logistic Regression, LinearSVC, Random Forest, Gaussian Naive Bayes ve K-En Yakın Komşu gibi sınıflandırıcılar ile yazar ve tür tahmini için kullanılmıştır. Ayrıca bu tezde, klasik Kosinüs Benzerliği

yaklaşımının ötesine geçen, vektör merkezli benzerlik ölçümüne dayalı bir yöntem benimsenmiştir. Her yazar ve kategori için eğitim verisinden bir “merkez vektör” çıkarılmış, test kitapları bu merkezlere olan açısal yakınlıklarına göre sınıflandırılmıştır. Bu yaklaşım, modelin kavramsal benzerliklere daha duyarlı çalışmasını sağlamış ve yüksek doğruluk oranlarıyla başarısını ortaya koymuştur. Tezin ikinci aşamasında geliştirilen öneri sistemi, kullanıcı davranışına dayalı filtreleme yöntemlerinden farklı olarak metinsel içerik benzerliklerine odaklanmakta; böylece okuyucuya beğendiği yazarlarla benzer anlatım tarzı taşıyan yeni yazarlar önermektedir. Bu yöntem özellikle az tanınan yazarların görünürlüğünü artırma potansiyeli taşımaktadır. Bu çalışma, belge gömme tabanlı yöntemlerin dijital beşeri bilimler, yazar tahmini ve öneri sistemleri gibi alanlarda etkili bir biçimde kullanılabileceğini göstermektedir.

Anahtar Kelimeler: Doc2Vec, Makine Öğrenmesi, Doğal Dil İşleme, Tür Sınıflandırması, Metin Sınıflandırması, Yazar Tanıma

To my family…

ACKNOWLEDGEMENT

I would like to thank my thesis supervisor, Asst. Prof. Ayça Kumluca Topallı, for her valuable guidance, support, and feedback throughout this research process. Her contributions have played an important role in shaping the direction and clarity of this thesis.

I also extend my sincere thanks to my parents, Süheyla Özyurt and Erkan Özyurt, for their continuous encouragement and moral support. Their presence and motivation have been a great source of strength for me during this journey.

Lastly, I am grateful to the jury members, Assoc. Prof. Nalan Özkurt and Assoc. Prof. Mehmet Türkan, for their constructive comments and valuable evaluations.

TABLE OF CONTENTS

ABSTRACT iv
ÖZET vi
ACKNOWLEDGEMENT ix
TABLE OF CONTENTS x
LIST OF TABLES xiii
LIST OF FIGURES xv
LIST OF ABBREVIATIONS xvii
CHAPTER 1: INTRODUCTION 1
CHAPTER 2: LITERATURE REVIEW 3
2.1. Previous Research on Genre and Authorship Classification in Literary Texts 3
2.2. Pioneering and Recent Studies on Doc2Vec for Text Representation and Classification 4
2.3. Applications of Doc2Vec and Text Vectorization in Turkish-Language NLP: Evidence from IEEE and DergiPark Studies 6
2.4. Applications of NLP and Machine Learning in Turkish Theses (YÖKTez) 8
CHAPTER 3: TECHNICAL BACKGROUND 10
3.1. Artificial Intelligence 10
3.2. Machine Learning 10
3.3. Natural Language Processing (NLP) 11
3.4. Tokenization 12
3.5. Vectorization Techniques 13
3.5.1. Term Frequency-Inverse Document Frequency (TF-IDF) 14
3.5.2. Word2Vec 16
3.5.3. Doc2Vec 18
3.6. Cosine Similarity 19
3.7. Vector Visualization Techniques: Heatmap and t-SNE 20
3.7.1. Heatmap 21
3.7.2. t-SNE (t-distributed Stochastic Neighbor Embedding) 22
3.8. Performance Evaluation Metrics 22
3.8.1. Accuracy 22
3.8.2. Precision 23

3.8.3. Recall 23
3.8.4. F_1-Score 24
3.8.5. Confusion Matrix 25
CHAPTER 4: METHODOLOGY 26
4.1. System Architecture and Model Overview 26
4.2. Data Preprocessing 28
4.2.1. Text Formatting and File Naming 28
4.2.2. Text Cleaning and Preprocessing Steps 29
4.2.3. Word Cloud Visualization 31
4.2.4. Tokenization and Construction of the TaggedDocument Structure 32
4.3. Embedding Algorithm 33
4.4. Model Training and Classification 34
4.4.1. Centroid-Based Classification via Vector Similarity 34
4.4.2. Machine Learning-Based Classification Approach 35
4.5. Performance Evaluation 36
4.6. Visualization: Analysis of Doc2Vec Embeddings 37
4.6.1. Cosine Similarity Analysis via Heatmap 37
4.6.2. Two-Dimensional Projection via t-SNE 38
4.7. Train-Test Splitting 38
4.7.1. In-Sample Evaluation: Construction of the Training and Test Sets 39
4.7.2. Out-of-Sample Evaluation: Genre Classification for Unseen Authors 39
4.8. On the Role and Significance of Classification 40
4.9. Development Environment 40
CHAPTER 5: RESULTS AND DISCUSSION 41
5.1. Evaluation on In-Sample Authors 41
5.1.1. Dataset Overview 41
5.1.1.1. English Dataset 42
5.1.1.2. Turkish Dataset 53
5.2. Model Results and Analysis 63
5.2.1. Doc2Vec Configuration Overview 63
5.2.2. Heatmap Analysis 64
5.2.2.1. Heatmap Analysis - English Dataset 65
5.2.2.2. Heatmap Analysis - Turkish Dataset 67
5.2.3. Genre Visualization with t-SNE 68

5.2.3.1. English Dataset - Genre t-SNE 69
5.2.3.2. Turkish Dataset - Genre t-SNE 70
5.2.4. Author Visualization with t-SNE 71
5.2.4.1. English Dataset - Author t-SNE 72
5.2.4.2. Turkish Dataset - Author t-SNE 73
5.2.5. Genre Classification Results 74
5.2.5.1. With English Dataset 74
5.2.5.2. With Turkish Dataset 78
5.2.6. Author Classification Results 82
5.2.6.1. With English Dataset 82
5.2.6.2. With Turkish Dataset 86
5.2.7. General Observations and Insights 89
5.2.8. Out-of-Sample Evaluation 90
5.2.8.1. Genre Prediction - English (Unseen Authors) 95
5.2.8.2. Genre Prediction - Turkish (Unseen Authors) 98
5.2.8.3. Author Prediction - English (Unseen Authors) 102
5.2.8.4. Author Prediction - Turkish (Unseen Authors) 105
CHAPTER 6: CONCLUSION 108
6.1. Suggestions for Future Studies 109
REFERENCES 111

LIST OF TABLES

Table 1. English Books Dataset 42
Table 2. Turkish Books Dataset 53
Table 3. English Genre Classification Results (Machine Learning and Centroid-Based Cosine Similarity) 74
Table 4. Comparison of macro-averaged Accuracy, Precision, Recall, and F_1-score for English genre classification models 77
Table 5. Turkish Genre Classification Results (Machine Learning and Centroid-Based Cosine Similarity) 78
Table 6. Comparison of macro-averaged Accuracy, Precision, Recall, and F_1-score for Turkish genre classification models 80
Table 7. English Author Classification Results (Machine Learning and Centroid-Based Cosine Similarity) 82
Table 8. Comparison of macro-averaged Accuracy, Precision, Recall, and F_1-score for English author classification models 84
Table 9. Turkish Author Classification Results (Machine Learning and Centroid-Based Cosine Similarity) 86
Table 10. Comparison of macro-averaged Accuracy, Precision, Recall, and F_1-score for Turkish author classification models 88
Table 11. English Genre Classification Results - Out-of-Sample Evaluation (Machine Learning and Centroid-Based Cosine Similarity) 91
Table 12. Turkish Genre Classification Results - Out-of-Sample Evaluation (Machine Learning and Centroid-Based Cosine Similarity) 93
Table 13. English Genre Classification Results - Out-of-Sample Evaluation (Machine Learning and Centroid-Based Cosine Similarity) 95
Table 14. Comparison of macro-averaged Accuracy, Precision, Recall, and F_1-score for English genre classification models – Out-of-Sample Evaluation 97
Table 15. Turkish Genre Prediction - Unseen Authors (Machine Learning and Centroid-Based Cosine Similarity) 98
Table 16. Comparison of macro-averaged Accuracy, Precision, Recall, and F_1-score for Turkish genre classification models – Unseen Authors 100
Table 17. English Author Prediction - Unseen Authors (Machine Learning and

Centroid-Based Cosine Similarity) 102
Table 18. Turkish Author Prediction - Unseen Authors (Machine Learning and Cosine Similarity) 105

LIST OF FIGURES

Figure 1. Sample document set composed of four short English sentences used for TF-IDF computation. 16
Figure 2. TF-IDF score matrix calculated from the sample sentences shown in Figure 1. Each row represents a document, and each column corresponds to a unique word. The cell values indicate the relative importance of each word in its respective document. 16
Figure 3. Two-dimensional projection of the words most semantically similar to “atatürk” from a Word2Vec model trained on a Turkish literary corpus. 17
Figure 4. Heatmap representation of the pairwise cosine similarity scores between authors’ document vectors, based on models trained using Doc2Vec. 21
Figure 5. Integrated system architecture for genre classification and author similarity detection 27
Figure 6. Sample raw text (before cleaning) 29
Figure 7. Sample text segmented into clean sentences 30
Figure 8. Sample TaggedDocument structure for Doc2Vec input 31
Figure 9. Word cloud visualization based on the writings of Sabahattin Ali 32
Figure 10. 200-dimensional vector output for the book SA8_ExampleText 33
Figure 11. Cosine similarity heatmap of English training set book vectors. (This heatmap visualizes the semantic and stylistic similarities between English books in the training set. Darker regions indicate higher similarity, often observed between books by the same author.) 65
Figure 12. Cosine similarity heatmap of Turkish training set book vectors. (This heatmap displays the cosine similarity scores between Turkish books in the training set. Clusters with stronger color intensities reflect semantic or stylistic similarity, frequently aligned with author-based groupings.) 67
Figure 13. t-SNE visualization of English book vectors grouped by genres. Each point represents a book, and each triangle denotes the centroid of a genre. 69
Figure 14. t-SNE visualization of Turkish book vectors grouped by genres. Each point represents a book, and each triangle denotes the centroid of a genre. 70
Figure 15. t-SNE visualization of English book vectors grouped by author identities. Each point represents a book, and each triangle denotes the centroid of an author... 72

Figure 16. t-SNE visualization of Turkish book vectors grouped by author identities. Each point represents a book, and each triangle denotes the centroid of an author... 73

LIST OF ABBREVIATIONS

AC : Agatha Christie

ACD : Arthur Conan Doyle

AK : Ayşe Kulin

AKG : Anna Katharine Green

AL : Andrew Lang

AN : Aziz Nesin

AT : Ayfer Tunç

AU : Ahmet Ümit

BIOG : Biography Memoir

CD : Charles Dickens

CLAS : Classic Novel

CM : Christopher Marlowe

DC : Doğan Cüceloğlu

DETE : Detective Novel

DRAM : Drama Tiyatro

EAP : Edgar Allan Poe

EEDS : E E Doc Smith

ED : Emily Dickinson

EG : Emile Gaboriau

ESSA : Essay Culture

ES : Elif Şafak

FD : Fyodor Dostoevsky

GBS : George Bernard Shaw

HGW : H G Wells

HI : Halil İnalcık

HI1 : Henrik Ibsen

HISF : Historical Fiction

HIST : History

HRG : Hüseyin Rahmi Gürpınar

IG : İsmail Gezgin

IO : İlber Ortaylı

IP : İskender Pala

JA : Jane Austen

JV : Jules Verne

KIDS : Children and Young Adult

LC : Lewis Carroll

LFB : L Frank Baum

LMA : Louisa May Alcott

MODN : Modern Novel

MR : Mehmet Rauf

MS : Mary Shelley

NY : Nermin Yıldırım

OA : Oğuz Atay

OP : Orhan Pamuk

OW : Oscar Wilde

PERS : Personal Development

POET : Poetry

RAF : R Austin Freeman

RF : Robert Frost

RLS : Robert Louis Stevenson

RK : Rudyard Kipling

SA : Sabahattin Ali

SATI : Satire Humour

SB : Samed Behrengi

SCFI : Science Fiction

SHOR : Short Story

SOCI : Social Critique

TB : Tarık Buğra

VH : Victor Hugo

WC : Wilkie Collins

WS : William Shakespeare

YK : Yaşar Kemal

ZL : Zülfü Livaneli

CHAPTER 1: INTRODUCTION

The rapid progress in Natural Language Processing (NLP) has facilitated the conversion of linguistic elements into compact numerical forms, commonly referred to as embedding vectors. This capability has significantly advanced the mathematical analysis of texts and made it possible to perform tasks such as question answering, sentence completion, machine translation, and sentiment analysis in an automated manner.

Genre, as defined by Merriam-Webster (2024), refers to

> *"a category of artistic, musical, or literary composition characterized by a particular style, form, or content."*

Traditionally, determining the genre of a literary work has been a manual process performed by editors or literary scholars, requiring the full reading and subjective evaluation of the text. However, modern NLP technologies offer the potential to identify the genre or stylistic similarities of newly published works without the need for such exhaustive manual review. This technological capacity enables an immediate understanding of whether a new work resembles well-known genres or authors, serving as a useful preliminary classification tool.

Readers often tend to choose books that resemble the works of authors they already admire. This preference is shaped by a combination of aesthetic orientation, trust in familiar names, and a tendency to seek familiar themes or narrative styles. A recent study by Sakal and Proulx (2023) analyzed Goodreads data and revealed that readers frequently prefer books similar in genre to those they have previously enjoyed. Another study by Vakkari and Mikkonen (2020) demonstrated that readers with strong aesthetic orientations and greater familiarity with specific authors not only exert less effort in the book selection process but also report higher satisfaction with their chosen books.

These findings indicate that readers might hesitate to engage with unfamiliar authors, but would be more inclined to do so if informed that a given book bears stylistic or thematic similarities to works by authors they already appreciate. In this context,

automatic genre identification and author-style matching systems have the potential to enhance literary discovery and reader engagement.

This thesis investigates whether it is possible to automatically determine the genre of a new literary work and to identify which well-known author's style it most closely resembles, using NLP and machine learning (ML) techniques. To address these research questions, two novel models have been proposed: the Genre-Finder and the Author-Matcher. In both approaches, a collection of books written by various authors across different genres was first converted into dense document embeddings. These vectors were then analyzed through similarity-based methods and several ML algorithms to predict the genre or author of unseen literary works. The proposed methods have been applied to both Turkish and English texts.

As far as we are aware, earlier research has not explored literary texts by simultaneously classifying their genres and determining their stylistic resemblance to prominent authors using embedding-driven methods. Therefore, the system proposed in this research presents an original advancement in the field.

The remainder of this thesis is structured to provide a coherent and systematic progression through the research. Chapter 2 reviews the existing body of literature related to genre classification and author-style modeling, highlighting the gap this study aims to fill. Chapter 3 outlines the fundamental concepts underlying artificial intelligence, machine learning, natural language processing, and vectorization, which serve as the theoretical basis for the proposed approach. Chapter 4 introduces the methodology, detailing the processes of data preparation, embedding generation, and the classification strategies employed. Chapter 5 presents the results of the experiments, including both quantitative evaluations and visual analyses of the models' performance. Chapter 6 wraps up the study with a summary of the core results and presents recommendations for future academic exploration.

CHAPTER 2: LITERATURE REVIEW

The increasing integration of Natural Language Processing (NLP) and Machine Learning (ML) techniques into the field of literary analysis has enabled researchers to examine textual data in more systematic and scalable ways. This chapter provides an overview of key academic studies focused on genre classification and authorship attribution in literary texts. Additionally, it explores the evolution of document embedding techniques, with a particular emphasis on the development and application of the Doc2Vec algorithm in text classification tasks.

2.1. Previous Research on Genre and Authorship Classification in Literary Texts

Academic studies aiming to classify the genres of literary texts such as novels, short stories, poems, and essays, or thematic categories like detective fiction, romance, and science fiction using NLP techniques are available in the literature. These studies typically employ methods such as text classification, stylistic analysis, topic modeling, and machine learning.

Kuzman and Ljubešić (2025) conducted a comprehensive survey that examined genre classification methods used in literary text analysis. They compared the performance of shallow neural networks such as fastText with Transformer-based models like XLM-RoBERTa, highlighting the superior performance of Transformer models, particularly on small datasets and in multilingual contexts. Similarly, Goyal and Prakash (2022) compared a range of machine learning and deep learning approaches on texts from Project Gutenberg, reporting high classification accuracy using models such as BERT, LSTM, and CNN.

Monte-Serrat, Machado and Ruiz (2021) applied random forests and tools such as LIWC to classify genres in Portuguese literary texts. Gupta, Agarwal and Jain (2019) used both traditional and deep learning models to classify book genres based on titles and short text excerpts. Stamatatos, Fakotakis and Kokkinakis (2000) performed genre and author classification on Modern Greek texts and demonstrated the effectiveness of stylometric features in such tasks.

In another line of research, Shavrina focused on the internal textual features of Russian web corpus “Taiga” to develop a genre classification approach. Her work emphasized the value of intrinsic linguistic characteristics, such as word usage, sentence structure, and morphological and syntactic features, in genre differentiation (Pinto Cerqueira et al., 2023).

Beyond genre identification, a number of academic studies have aimed to predict the authors of literary texts using machine learning. For instance, Bauersfeld et al. (2023) employed a Transformer-based deep learning model to attribute anonymous academic papers to their likely authors using content and bibliography data. The model, trained on approximately two million articles from arXiv, achieved accuracy rates as high as 73 percent for subsets with 2000 different authors.

da Rocha Bartolomei and Drummond (2020) identified authors in Brazilian literary texts using various machine learning techniques, achieving high accuracy by detecting stylistic differences among texts. Hu, Wang and Wu (2014) conducted a quantitative study on the classical Chinese novel Dream of the Red Chamber, applying support vector machines and other methods to suggest that the novel may have multiple authors based on chapter-wise analysis.

Fedotova et al. (2022) used models such as SVM, fastText, and deep neural networks to perform authorship attribution in both social media and literary texts in Russian, reporting high prediction accuracy. Walkowiak (2023) applied named entity masking and MaxLogit-based sequence classification to perform authorship attribution in Polish literature, achieving strong results even with short texts.

2.2. Pioneering and Recent Studies on Doc2Vec for Text Representation and Classification

One of the foundational studies that introduced the Doc2Vec method was presented by Le and Mikolov (2014). In their influential work titled *Distributed Representations of Sentences and Documents*, the authors addressed key limitations of the Bag-of-Words approach namely, its disregard for word order and its production of sparse, high-dimensional vectors. They proposed two variants of the Paragraph Vector model:

PV-DM (Distributed Memory) and PV-DBOW (Distributed Bag of Words), which generate dense fixed-length representations for paragraphs and documents. These models were trained using backpropagation and stochastic gradient descent. Evaluations conducted on the Stanford Sentiment Treebank and IMDB sentiment classification tasks showed that Doc2Vec significantly outperformed traditional methods such as BoW, recursive models, and TF-IDF with bigrams, reducing error rates by 15-32% depending on the task.

Abubakar et al. (2022) provided a comparative review of four common text vectorization techniques Bag-of-Words (BoW), TF-IDF, Word2Vec, and Doc2Vec within the context of sentiment classification. The study highlighted that while TF-IDF performed well on structured datasets, the combination of TF-IDF and Word2Vec yielded the highest classification accuracy. The review used standard evaluation metrics including accuracy, precision, recall, and F_1-score.

Another notable application of Doc2Vec was demonstrated by Budiarto et al. (2020), who employed it for unsupervised topic modeling on a large corpus of over one million Indonesian news articles. A representative subset of 28,402 texts was converted into document embeddings using Doc2Vec, which were then grouped into 11 categories using cosine-based spherical clustering. The system's performance was measured using "Best-1", "Best-3", and "Best-5" match accuracy, reaching 87.30% in Best-5. These results showed that while supervised LSTM models still led in top-1 accuracy, the Doc2Vec + clustering approach was highly effective for grouping large unlabeled datasets into coherent topic categories without human intervention.

Kim et al. (2019) explored document classification in low-resource settings using their Multi-Co-Training (MCT) strategy, which integrated three distinct document representations: TF-IDF, LDA, and Doc2Vec. Each representation was used to train separate classifiers, which then collaboratively improved the final model through co-training. Tested on widely-used datasets such as Reuters-21578, 20 Newsgroups, Ohsumed, Reuters-50-50, and SemEval with varying label ratios from 2% to 50%, the MCT method consistently achieved higher accuracy and stability compared to other baseline models.

2.3. Applications of Doc2Vec and Text Vectorization in Turkish-Language NLP: Evidence from IEEE and DergiPark Studies

Several peer-reviewed studies in IEEE and DergiPark databases have applied Doc2Vec and related embedding methods to Turkish-language texts for various classification and similarity tasks. These works contribute to the growing body of literature on document-level vectorization in underrepresented languages, especially Turkish.

In a study by Torun and İnner (2022), over 28,000 Turkish news articles were vectorized using both TF-IDF and Doc2Vec representations of varying dimensions (25-100), and cosine similarity combined with the Annoy library was used for nearest neighbor detection. While TF-IDF excelled in exact match scenarios, 40-60 dimensional Doc2Vec vectors proved more memory-efficient and effective for semantic matching.

Jabrayilzade et al. (2020) presented a multi-label classification dataset for Turkish movie genres based on 20,503 films. Each film was represented using both Latent Dirichlet Allocation (LDA) and Doc2Vec, and classification was performed using fully connected neural networks. The results showed that while Doc2Vec performed slightly better in Exact Match accuracy for short texts, LDA outperformed it in F_1-scores across both short and long texts.

Çoban and Karabey (2016) explored genre classification in Turkish song lyrics using TF-IDF, Word2Vec, and Doc2Vec (DM and DBoW). Their findings indicated that Bag-of-Words outperformed all others in accuracy (67.28%), while Doc2Vec yielded the lowest accuracy (38.13%), especially as vector dimensions increased.

Sarı and Özbayoğlu (2018) utilized PV-DM and PV-DBOW to classify 20,000 Turkish newspaper columns written by 20 different authors. PV-DM, which considers word order, consistently outperformed PV-DBOW, particularly in texts with distinct stylistic features.

Deniz et al. (2019) vectorized 799 Turkish emails (400 non-spam, 399 spam) using Doc2Vec models and compared classifiers such as SVM, Naive Bayes, and Random Forest. The combination of DM and DBoW with SVM yielded the highest accuracy, confirming Doc2Vec's utility for spam detection in Turkish.

Doğru et al. (2021) compared Doc2Vec (PV-DM and PV-DBOW) models with CNN and traditional classifiers on both Turkish and English news corpora. When combined with proper preprocessing, the Doc2Vec+CNN approach achieved 94.17% accuracy on Turkish data and 96.41% on English, outperforming other techniques.

Çelenli et al. (2018) tested Doc2Vec embeddings on Turkish texts using several classifiers. SVM consistently delivered the highest accuracy, followed by Logistic Regression and Random Forest, validating the superiority of Doc2Vec over traditional Bag-of-Words models in low-dimensional representations.

Erdinç and Güran (2019) explored semi-supervised categorization with 6.7 million Turkish news articles using Word2Vec, Doc2Vec, and FastText embeddings. Their results showed that FastText, especially with character-level n-grams and punctuation cleaning, achieved the highest F_1-score (0.780).

Bilgin and Şentürk (2017) trained semi-supervised Doc2Vec models on both Turkish and English tweets to improve sentiment classification. The DBoW model outperformed DM, and the larger, more balanced English dataset achieved higher accuracy, illustrating the importance of corpus quality in Doc2Vec-based applications.

Kınık and Güran (2021) enhanced Turkish text categorization by integrating phrase modeling (frequent bigrams/trigrams and Wikipedia phrase extraction) into TF-IDF and Doc2Vec representations. Combined with ensemble classifiers, this approach significantly improved classification accuracy, up to 95% with Doc2Vec.

Ayanoğlu et al. (2023) developed a system to classify customer complaints in Turkish using Word2Vec+CNN+LSTM, which achieved the highest F_1-score (95%) for single-label examples, outperforming BERT and other baselines. Doc2Vec was not directly applied in this study but contributes contextually to the Turkish NLP classification landscape.

Aydoğan and Karcı (2019) used a large-scale Turkish corpus (~10.5 billion words) to compare Word2Vec and GloVe embeddings, evaluating word similarities with cosine and Euclidean distances. Word2Vec, particularly CBOW, outperformed GloVe in both speed and accuracy.

2.4. Applications of NLP and Machine Learning in Turkish Theses (YÖKTez)

Several master's theses published in Turkey provide valuable insight into the application of natural language processing (NLP) and machine learning (ML) techniques for author identification, genre classification, psychological state prediction, and education-focused text analysis.

Samet Kaya (2018) conducted a study on identifying book authors using NLP techniques. A dataset of 120 books from 20 different authors was processed to remove noise and standardized through preprocessing. Character n-grams (bi-, tri-, and quadri-gram) were extracted, and the classification was performed using a Naive Bayes classifier. The highest accuracy of 82% was achieved with the quadri-gram approach.

Buse Demir (2024) aimed to predict individuals' psychological distress levels using narrative texts from Reddit. The texts were converted to numerical vectors using TF-IDF and Word2Vec, and classification was performed using Naive Bayes, SVM, KNN, and XGBoost, focusing on conditions such as depression and social anxiety.

Emre Mumcuoğlu (2022) developed a system to predict judicial decisions from Turkish Constitutional and regional court rulings. The texts were vectorized using TF-IDF and Word2Vec, and classifiers including decision trees, random forest, SVM, and deep learning models such as GRU and bi-LSTM with attention mechanisms were employed.

Emre YÖNET (2023) classified Turkish poems into thematic and stylistic categories such as "love" or "longing." A dataset of 4,198 poems was vectorized using character-level n-grams and TF-IDF. Six classifiers were compared, and Random Forest and SVM performed best with F_1-scores ranging from 88% to 90%.

Serkan Korkmaz (2021) focused on poet identification through morphological analysis of 2,805 poems by 75 poets. Root and affix separation was performed to extract linguistic features, which were used to train ML models for author prediction.

Tuğba Yegen (2023) applied NLP tools to analyze science textbooks for primary and middle school levels. Using NKod (Zemberek-based), both word and lemma frequency lists were generated, offering insights for comparative linguistic and pedagogical evaluation.

Ayhkan Baghirov (2024) compared ML methods for poet identification on a large corpus of 120,976 poems from 18,499 poets. Preprocessing included morphological analysis using Zemberek. TF-IDF was used for vectorization, and top-performing classifiers were Logistic Regression, Stochastic Gradient Descent, and Extra Trees, achieving up to 89% accuracy.

Ibrahim Doğan (2023) combined stylometric and NLP-based methods for matching Turkish texts to their authors. News articles were transformed into TF-IDF and BoW vectors. Seven classifiers were evaluated, and the use of rich morphological features tailored to the agglutinative nature of Turkish demonstrated high efficacy in author attribution and plagiarism detection.

Although numerous studies have been conducted using NLP and machine learning techniques to classify literary genres and predict authorship (including several in the context of Turkish texts) the reviewed literature does not appear to include any method that jointly determines both the genre and the most stylistically similar well-known author of a newly published or lesser-known literary work. Most existing studies tend to focus on either genre or authorship independently. Furthermore, while genre classification and authorship attribution have been explored separately using Turkish corpora, no prior study has proposed a unified, automated framework that simultaneously performs both tasks. In particular, the problem of identifying the literary genre and the closest stylistic match for a work written by a new or relatively unknown author has not been directly addressed. In this regard, the approach outlined in this thesis presents a novel contribution to the field by offering an integrated NLP-based method tailored to Turkish literary data. Such a system has the potential to benefit not only computational literary analysis but also stakeholders in the publishing industry and general readership.

CHAPTER 3: TECHNICAL BACKGROUND

3.1. Artificial Intelligence

Artificial Intelligence (AI) refers to a multidisciplinary domain that seeks to emulate key aspects of human cognition, including learning, reasoning, solving problems, perceiving, and comprehending language, via computational systems. The foundational concept of AI was initially proposed by John McCarthy during the Dartmouth Conference of 1956, an event widely acknowledged as the formal inception of AI as a research discipline (McCarthy et al., 2006). Since its introduction, AI has progressively impacted diverse fields such as computer science, psychology, linguistics, and engineering, and has become a driving force in today's technological evolution.

In contrast to conventional rule-based systems, AI technologies are designed to derive insights from data and make independent decisions. Rather than relying on fixed logic, they detect patterns and connections within expansive datasets and adjust dynamically to varying environments. This adaptability is largely enabled by subdomains like machine learning, deep learning, and natural language processing.

Today, AI technologies are widespread across a range of industries, including healthcare, finance, education, defense, and manufacturing. Particularly in language-centric tasks, AI contributes to areas such as text classification, recommendation algorithms, machine translation, and speech recognition, showcasing its effectiveness in handling and interpreting linguistic data.

3.2. Machine Learning

Machine Learning (ML) is a domain within artificial intelligence that empowers computational systems to carry out designated tasks by acquiring knowledge from historical data, without requiring direct programming for each action. In contrast to conventional rule-based frameworks that depend on fixed instructions, ML algorithms

enhance their capabilities through experiential learning, recognizing underlying patterns and correlations in data and applying these insights to new, unseen situations.

The fundamental principle behind machine learning algorithms involves constructing mathematical representations of the relationships between input and output variables. During the learning process, a model interacts with a dataset, referred to as the training set, by fine-tuning its internal parameters to reflect the regularities it identifies in the data. The model's ability to generalize is then assessed using a distinct test set composed of data it has not encountered previously. This methodology marks a shift away from rigid programming paradigms toward systems that evolve dynamically through data exposure.

Machine learning approaches are commonly divided into three primary categories: supervised learning, unsupervised learning, and reinforcement learning. In supervised learning, models are developed using labeled datasets in which each input corresponds to a predefined output, with algorithms such as Logistic Regression, Random Forest, and Support Vector Machines (SVM) frequently employed. Conversely, unsupervised learning engages with unlabeled data in order to discover latent structures or clusters through methods like clustering and dimensionality reduction, utilizing techniques such as Principal Component Analysis (PCA) and t-distributed Stochastic Neighbor Embedding (t-SNE). Reinforcement learning, in contrast, involves agents that iteratively adapt by interacting with their environment and modifying behavior in response to feedback, which often takes the form of rewards or penalties. This strategy has proven effective in fields like autonomous robotics and strategic game development.

3.3. Natural Language Processing (NLP)

Natural Language Processing (NLP) is a branch of artificial intelligence dedicated to enabling machines to comprehend and manipulate human language in both its spoken and written forms. By converting the inherent complexity of linguistic data into structured, numerical representations, NLP facilitates computational analysis of language. This discipline serves as a crucial bridge between human communication and machine interpretation, allowing algorithms to carry out advanced tasks such as

summarizing text, translating content, analyzing sentiment, and recognizing speech with growing precision and fluency.

The NLP workflow is generally composed of several key preprocessing stages. These include text normalization procedures such as the removal of punctuation, special symbols, and control characters; tokenization, which divides the text into individual words, sentences or smaller linguistic units; and the application of lowercase transformation to standardize case. Additional steps involve eliminating stop words (terms with limited semantic value such as "and" or "the") as well as employing stemming and lemmatization techniques that convert words to their root or canonical forms. The final stage typically involves vectorization, which encodes text into numerical formats using approaches such as Term Frequency-Inverse Document Frequency (TF-IDF), Word2Vec, or Doc2Vec.

These preprocessing techniques are vital for preparing textual data for statistical modeling and machine learning applications. Once the input has been cleaned and systematically organized, it can support a wide array of NLP-related tasks. Among these, vectorization is especially important because it transforms linguistic elements into dense numerical embeddings, which downstream models can effectively utilize. This enables higher-level analysis such as document classification, text clustering, and semantic similarity detection.

3.4. Tokenization

Tokenization represents a foundational and essential component within the natural language processing (NLP) workflow. It refers to the process of segmenting raw textual input into smaller, semantically meaningful elements, most often individual words, though sometimes entire sentences or even finer-grained subword fragments. These segments, commonly referred to as tokens, serve as discrete informational units that enable further computational processing. They constitute the structural groundwork for subsequent analytical tasks such as classification, content analysis, and vector-based representation. By converting unstructured linguistic input into clearly defined and manageable segments, tokenization facilitates the application of

statistical techniques and machine learning models to language data, thereby making the content computationally actionable.

Tokenization often involves several concurrent preprocessing steps, including the separation or removal of punctuation marks, elimination of special characters (such as @, #, %), and normalization of excessive spacing. Optionally, the entire text may be converted to lowercase to maintain consistency. In languages with rich morphology (such as Turkish, German, or Finnish), tokenization can be more complex and may require language-specific morphological rules or external libraries for proper segmentation.

3.5. Vectorization Techniques

In natural language processing (NLP), raw text cannot be directly used by numerical algorithms. Before textual data can be utilized as input to machine learning models, it must first be converted into a numerical format. This transformation process is referred to as vectorization.

Vectorization involves representing a text as a vector of fixed dimensions. These vectors are designed to capture various aspects of the text, such as semantic content, statistical distributions, or contextual relationships. Once converted, textual data can be processed in numerical space for a variety of tasks including classification, clustering, and similarity measurement.

Among the various strategies available for transforming textual data into vectorized formats, this thesis places particular emphasis on three prominent techniques. The first is Term Frequency-Inverse Document Frequency (TF-IDF), a statistical approach that determines the significance of individual words by measuring their frequency in a specific document relative to their distribution across an entire corpus. TF-IDF is widely used as a foundational model for text representation. The second technique is Word2Vec, an embedding method that generates high-dimensional vector representations of words by capturing their contextual usage within large datasets. The third approach, Doc2Vec, builds upon the Word2Vec architecture by producing embeddings that represent entire documents rather than isolated words, thereby

encapsulating broader semantic content. This final method constitutes the primary vectorization strategy utilized in the current research.

The subsequent sections of this thesis provide a detailed exploration of each of these techniques. It is essential to recognize that vectorization methods are not universally optimal; their effectiveness varies according to the specific characteristics of the textual data and the objectives of the analytical task. Given that this study is concerned with representing extended narrative texts, such as books. Particular attention has been given to models that operate at the document level rather than at the word level, with Doc2Vec receiving special focus due to its alignment with the study's objectives.

3.5.1. Term Frequency-Inverse Document Frequency (TF-IDF)

Term Frequency-Inverse Document Frequency (TF-IDF) is a well-established vectorization method frequently employed in both text mining and natural language processing (NLP) tasks. This technique assigns significance to a term by evaluating how often it appears within a given document while simultaneously accounting for how commonly the same term occurs across an entire corpus. By doing so, TF-IDF downplays the weight of frequently occurring yet semantically insignificant words (such as "and," "the," or "this") and amplifies the relevance of less common but more meaningful terms.

The foundational premise of TF-IDF rests on two key principles. First, the more often a term appears within a single document, the more likely it is to hold contextual significance for that specific text. This aspect is quantified by the Term Frequency (TF) metric, which is calculated using the formula presented in Eq. 1:

$$TF(t,d) = \frac{Number\ of\ times\ term\ t\ appears\ in\ document\ d}{Total\ number\ of\ terms\ in\ document\ d} \quad (1)$$

Second, if a word appears in many documents, it is less useful for distinguishing between them. This is addressed by the Inverse Document Frequency (IDF), defined in Eq. 2:

$$IDF(t) = \log\left(\frac{N}{n_t}\right) \quad (2)$$

The final TF-IDF score is obtained by combining both components as in Eq. 3:

$$\text{TF-IDF}(t, d) = \text{TF}(t, d) \times \text{IDF}(t) \quad (3)$$

To illustrate this concept, a basic experiment was conducted using a small document set composed of four short sentences. These documents are visualized in Figure 1.

Each sentence underwent tokenization and preprocessing in order to compute TF-IDF values corresponding to individual terms across the document set. These computed values are illustrated in Figure 2, in which each row is associated with a specific document, and each column denotes a distinct term. The numerical entries in the matrix reflect the contextual relevance of each term with respect to its originating document.

For example, in Document 1, the term future receives a relatively high TF-IDF score of 0.523 because it appears exclusively in that document, making it highly distinctive. Conversely, the word is, which occurs in multiple documents, has a much lower score due to its lower inverse document frequency (IDF) value.

This example clearly demonstrates how TF-IDF effectively highlights distinctive vocabulary and suppresses common terms, thereby producing meaningful numerical representations of text that are suitable for machine learning algorithms.

```
documents = [
    "Artificial intelligence is the future.",
    "Machine learning is a part of artificial intelligence.",
    "Natural language processing helps computers understand text.",
    "TF-IDF is a feature extraction method in NLP."
]
```

Figure 1. Sample document set composed of four short English sentences used for TF-IDF computation.

	artificial	computers	extraction	feature	future	helps	idf	in	intelligence	is	language	learning	machine	method	natural	nlp	of	part	processing	text	tf	the	understand
Document 1	0.413	0.000	0.000	0.000	0.523	0.000	0.000	0.000	0.413	0.334	0.000	0.000	0.000	0.000	0.000	0.000	0.000	0.000	0.000	0.000	0.000	0.523	0.000
Document 2	0.332	0.000	0.000	0.000	0.000	0.000	0.000	0.000	0.332	0.269	0.000	0.421	0.421	0.000	0.000	0.000	0.421	0.421	0.000	0.000	0.000	0.000	0.000
Document 3	0.000	0.378	0.000	0.000	0.000	0.378	0.000	0.000	0.000	0.000	0.378	0.000	0.000	0.000	0.378	0.000	0.000	0.000	0.378	0.378	0.000	0.000	0.378
Document 4	0.000	0.000	0.367	0.367	0.000	0.000	0.367	0.367	0.000	0.235	0.000	0.000	0.000	0.367	0.000	0.367	0.000	0.000	0.000	0.000	0.367	0.000	0.000

Figure 2. TF-IDF score matrix calculated from the sample sentences shown in Figure 1. Each row represents a document, and each column corresponds to a unique word. The cell values indicate the relative importance of each word in its respective document.

3.5.2. Word2Vec

Word2Vec is a widely recognized embedding method within the field of natural language processing (NLP) that encodes words into continuous vector representations. Initially introduced by Mikolov et al. (2013), this model uncovers semantic relationships by analyzing the surrounding textual contexts in which words occur. As a result, it can place semantically related words near one another within a high-dimensional vector space.

This method operates on the foundations of distributional semantics, which asserts that words found in similar contexts typically share similar meanings. Word2Vec employs two primary architectural approaches: the Continuous Bag of Words (CBOW) model, which infers a target word from its surrounding words, and the Skip-Gram model, which predicts contextual words based on a given target word.

To interpret and visualize the internal structure of these vectorized word representations, dimensionality reduction techniques such as t-SNE (t-distributed

Stochastic Neighbor Embedding) are frequently utilized. t-SNE translates high-dimensional embeddings into a two-dimensional plane, maintaining the relative proximities between vectors. This process facilitates the visualization of semantic groupings learned by the model.

Figure 3 shows an example output of a Word2Vec model trained on a Turkish corpus composed of publicly available literary and historical texts, including novels, essays, and biographies. When queried with the word “atatürk”, the model returns semantically related words such as “dönemi” (period), “politikası” (policy), “ilkeleri” (principles), and “inkılap” (reform), which cluster closely in the two-dimensional space.

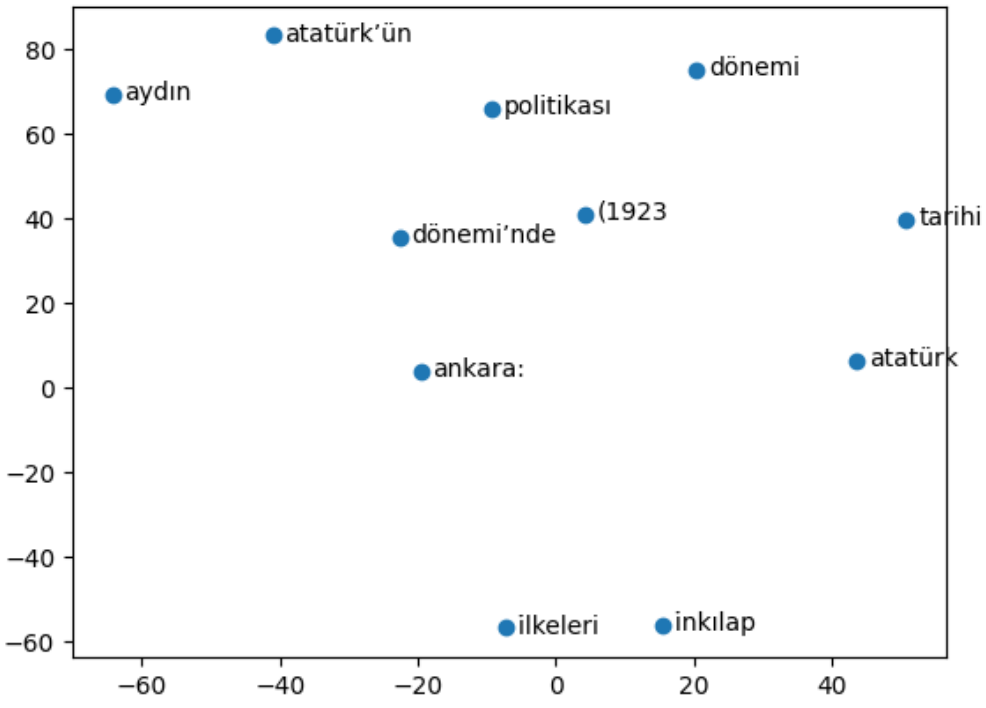

Figure 3. Two-dimensional projection of the words most semantically similar to “atatürk” from a Word2Vec model trained on a Turkish literary corpus.

This visualization illustrates how Word2Vec captures contextual meaning and organizes semantically similar words in coherent regions of the vector space, enabling deeper linguistic analysis and machine learning applications.

3.5.3. Doc2Vec

Doc2Vec builds upon the foundation laid by the Word2Vec model and was introduced by Mikolov et al. (2014) to enable the creation of fixed-length vector representations for entire documents, rather than just for individual words. While Word2Vec embeds semantically related terms near one another in a continuous vector space, Doc2Vec extends this idea to longer textual segments (such as paragraphs or full documents) thereby capturing not only localized semantic cues, but also broader contextual structures present within the text.

The core mechanism of Doc2Vec involves assigning a distinct vector to each document, which complements the word vectors used during training. These document vectors are refined in tandem with word vectors to predict surrounding words, allowing the model to encode both thematic and stylistic elements that span the entire document. This added vector serves as a contextual memory, enabling the model to generate document-level embeddings that reflect the input's overall tone, structure, and content.

There are two principal variants of the Doc2Vec framework: Distributed Memory (DM) and Distributed Bag of Words (DBOW). In the DM configuration, the document vector is combined with nearby word vectors to predict a target word, emphasizing the idea that documents maintain coherent semantic flows. In contrast, the DBOW model omits contextual word input entirely, relying solely on the document vector to infer the likely words, thereby prioritizing the encoding of global semantic information over local word relationships. This approach is especially suitable for long-form texts, such as literary works or academic articles, where preserving the structural and thematic integrity of the entire document is essential. Compared to traditional vectorization techniques like TF-IDF, which rely on term frequency statistics, Doc2Vec offers a more nuanced and semantically rich representation by learning from the distribution and sequence of words in context.

By transforming unstructured textual data into dense, fixed-length vectors, Doc2Vec serves as a foundational method for numerous downstream tasks in natural language processing, including document classification, clustering, similarity analysis, and recommendation systems.

3.6. Cosine Similarity

Cosine similarity is a commonly applied metric in natural language processing (NLP) and text mining, used to evaluate how similar two textual entities are. This approach begins by converting documents into numerical vector form and then determining the angular similarity between those vectors. Unlike metrics that rely on absolute distance, cosine similarity emphasizes the directionality of the vectors, not their magnitude.

From a mathematical standpoint, cosine similarity computes the cosine of the angle between two vectors, as outlined in Eq (4):

$$\text{Cosine Similarity} = \frac{\vec{A} \cdot \vec{B}}{|\vec{A}| \cdot |\vec{B}|} \tag{4}$$

In this formula $\vec{A}$ and $\vec{B}$ represent the document vectors being compared. The dot product $\vec{A} \cdot \vec{B}$ measures the directional similarity, while $|\vec{A}|$ and $|\vec{B}|$ denote the Euclidean norms (magnitudes) of the respective vectors.

Cosine similarity values theoretically range between -1 and 1. However, since text vectorization methods like TF-IDF, Word2Vec, or Doc2Vec typically yield non-negative values, the effective range is generally confined to 0-1. A value of 1 indicates perfect alignment (i.e., near-identical semantic content), whereas 0 implies orthogonality, meaning no shared similarity.

In the context of this research, cosine similarity served as a baseline technique for both author attribution and genre classification tasks. In this thesis, we refer to this approach as Centroid-Based Cosine Similarity, though it can also be described as Clustering in a broader context, since it groups document vectors around representative centroids. To do this, a centroid vector was calculated for each genre or author in the training dataset by averaging the associated document vectors. Each document from the test dataset was then evaluated against these centroids, and the genre with the highest similarity score was assigned as the prediction.

This Centroid-Based Cosine Similarity strategy provides a simple, yet effective mechanism for categorization based solely on vector geometry, without relying on supervised learning models. Consequently, it serves as a valuable benchmark for assessing the effectiveness of more advanced machine learning techniques.

3.7. Vector Visualization Techniques: Heatmap and t-SNE

Interpreting vector representations in high-dimensional spaces, particularly those generated in natural language processing (NLP) contexts, can present significant analytical challenges. To enhance interpretability and support qualitative assessments of model behavior, reducing these vectors into a two-dimensional format proves crucial. Among the most frequently applied techniques for this purpose are the following:

3.7.1. Heatmap

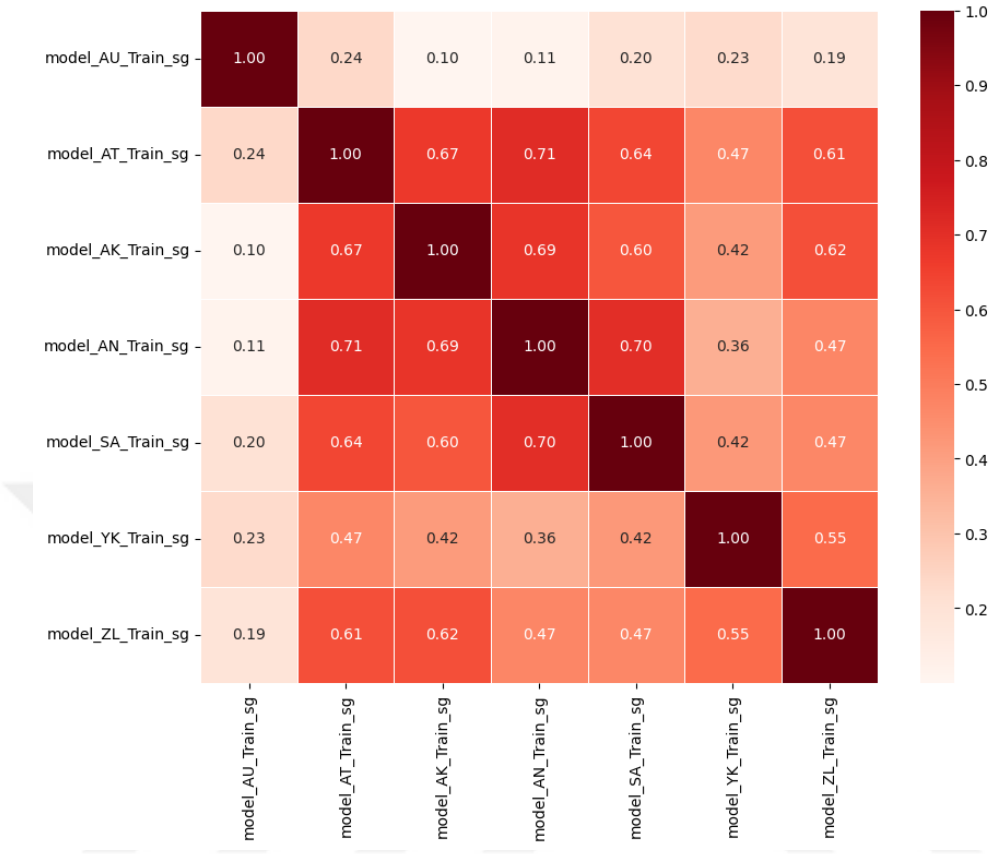

Figure 4. Heatmap representation of the pairwise cosine similarity scores between authors' document vectors, based on models trained using Doc2Vec.

This method presents a visual representation of the cosine similarity matrix computed between document vectors. As illustrated in Figure 4, color intensity indicates the degree of similarity: darker colors represent higher similarity values. Diagonal values (1.0) correspond to the similarity of each vector with itself, while off-diagonal regions display the similarity between different documents or authors. This visual format enables an intuitive comparison of semantic proximity within the vector space.

3.7.2. t-SNE (t-distributed Stochastic Neighbor Embedding)

t-SNE (t-distributed Stochastic Neighbor Embedding) is a technique used for reducing the dimensionality of data by transforming high-dimensional vectors into a two-dimensional representation. A practical example of this technique was previously illustrated in Figure 3, where the semantic neighborhood of the word “atatürk” was visualized using a two-dimensional projection derived from a Word2Vec model. This transformation retains the local neighborhood structure of the data, enabling clearer visualization of semantic groupings and cluster formations within the dataset.

3.8. Performance Evaluation Metrics

Evaluating the effectiveness of the developed models cannot rely exclusively on overall accuracy. As such, this study adopts a range of multidimensional performance indicators to assess the efficacy of both machine learning and similarity-based classification systems. These metrics provide a deeper insight into model performance, especially in cases involving class imbalance.

3.8.1. Accuracy

Accuracy measures the ratio of correct classifications to the total number of predictions. While it is commonly used as a general benchmark for evaluating model performance, this study gives particular attention to its limitations in imbalanced datasets, where accuracy alone may be misleading. The mathematical formulation of accuracy is presented as shown in Eq. (5):

$$\text{Accuracy} = \frac{TP+TN}{TP+TN+FP+FN} \tag{5}$$

In multi-class settings, overall accuracy is computed by dividing the total number of correct predictions across all classes by the total number of predictions, as shown in Eq. (6):

$$Accuracy = \frac{\sum_{i=1}^{K} TP_i}{Total\ Number\ of\ Samples} \quad (6)$$

3.8.2. Precision

Precision represents the proportion of correct positive predictions among all positive outcomes identified by the model. It assesses how well the model can distinguish and label only the truly relevant instances as positive. This metric becomes especially significant in contexts where false positives may result in costly consequences. The formula is given as shown in Eq. (7):

$$\text{Precision} = \frac{TP}{TP+FP} \quad (7)$$

In multi-class classification problems, this study employs macro-averaged precision, which evaluates each class separately and then averages the scores, treating all classes with equal importance regardless of their size. The macro-averaged version is shown in Eq. (8):

$$\text{Precision}_{\text{macro}} = \frac{1}{K}\sum_{i=1}^{K} \frac{TP_i}{TP_i+FP_i} \quad (8)$$

3.8.3. Recall

Recall, also known as sensitivity, measures the proportion of actual positive cases that are correctly identified by the model. It is especially important in scenarios where

failing to detect a positive case (i.e., producing a false negative) may lead to significant consequences. The corresponding formula is given in Eq. (9):

$$\text{Recall} = \frac{TP}{TP+FN} \tag{9}$$

As with precision, macro-averaged recall is utilized in this study to ensure that performance is assessed equally across all classes. This approach is particularly useful in datasets with class imbalance. The corresponding macro formula is shown in Eq. (10):

$$\text{Recall}_{\text{macro}} = \frac{1}{K}\sum_{i=1}^{K}\frac{TP_i}{TP_i+FN_i} \tag{10}$$

3.8.4. F_1-Score

The F_1-score represents the harmonic average of precision and recall, offering a unified measure that balances both metrics. It is particularly valuable in situations where it is essential to mitigate the impact of both false positives and false negatives. The formula used to compute the F_1-score is presented as shown in Eq. (11):

$$F_1 = 2 \times \frac{\text{Precision}\times\text{Recall}}{\text{Precision}+\text{Recall}} \tag{11}$$

In this study, macro-averaged F_1-score is also used to evaluate the overall performance of models across multiple classes in a balanced manner. This is computed by first calculating the F_1-score for each individual class and then averaging them equally, as presented in Eq. (12):

$$F_{1_{\text{macro}}} = \frac{1}{K}\sum_{i=1}^{K} 2 \times \frac{\text{Precision}_i\times\text{Recall}_i}{\text{Precision}_i+\text{Recall}_i} \tag{12}$$

3.8.5. Confusion Matrix

The confusion matrix offers a comprehensive overview of classification outcomes by category. It details the frequency of both correct and incorrect predictions by listing the number of true positives (TP), false positives (FP), true negatives (TN), and false negatives (FN). This analysis enables a deeper understanding of which specific classes posed the greatest difficulty for the models.

Each of the metrics discussed was calculated individually for every classification method used in the study. The corresponding results were presented as both tables and visual figures in Chapter 5. This facilitated a well-rounded comparison of model performance, highlighting not only the top-performing algorithms but also the particular categories that were most challenging to classify correctly.

CHAPTER 4: METHODOLOGY

This chapter presents the methodological approach adopted in the study, with a focus on the design of the proposed classification system and the techniques applied throughout the modeling process. It explains how literary texts are converted into numerical representations, introduces the structure of the dual-model framework, and describes the procedures used for both genre classification and authorship attribution. Each step of the pipeline is discussed in detail, including preprocessing, vectorization, and performance evaluation, to provide a clear and reproducible methodology.

4.1. System Architecture and Model Overview

This study introduces a dual-layered system architecture designed to classify the literary genre of a book and identify its most stylistically similar author. The architecture is built upon Natural Language Processing (NLP) and Machine Learning (ML) techniques, relying on semantic representations of texts generated via the Doc2Vec algorithm. The overall goal is to analyze literary texts in Turkish and English using both content-based and stylistic cues derived directly from their full content.

The proposed system comprises two core models: the Genre-Finder, which determines the genre of a text (e.g., detective novel, historical fiction, biography), and the Author-Matcher, which estimates which well-known author the input text stylistically resembles. These two models operate in parallel and are implemented using a shared document embedding infrastructure.

At the preprocessing stage, all books in the dataset are tokenized, cleaned, and transformed into dense fixed-length vectors using the Doc2Vec algorithm. These vectors capture both thematic and stylistic nuances of the texts, forming the basis for downstream classification.

To evaluate each book, the system employs two complementary approaches: vector-based similarity using centroid comparisons and supervised machine learning models. In the first approach, the vectors of books belonging to a particular genre or author are averaged to compute representative centroids. New books are then compared to these

centroids using cosine similarity, and the closest genre or author is selected accordingly. In the second approach, the same document vectors serve as input to a set of supervised classification algorithms, including Logistic Regression, Linear Support Vector Classifier (LinearSVC), Random Forest, Gaussian Naive Bayes, and K-Nearest Neighbors (KNN), which produce probabilistic predictions based on learned patterns.

To ensure the system's reliability and generalizability, the dataset is initially partitioned into training and test sets. Performance is first evaluated on this split to establish baseline accuracy. In a subsequent phase, the models are evaluated on texts by previously unseen authors, simulating real-world scenarios in which a new book may not conform to any known stylistic profile.

The overall architecture of the proposed system, which encompasses preprocessing, embedding, dual-mode classification, and evaluation stages, is presented in Figure 5.

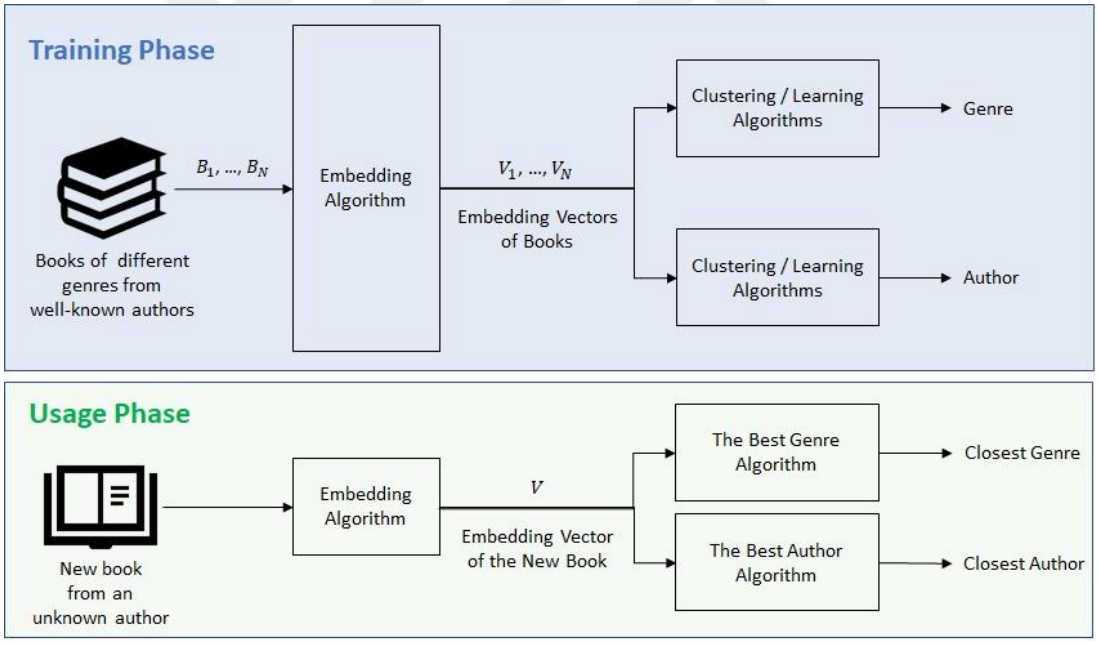

Figure 5. Integrated system architecture for genre classification and author similarity detection

This modular structure not only enables efficient classification of literary texts but also provides a robust platform for content-based recommendation. By analyzing each new text through both semantic similarity and trained statistical patterns, the system aims

to offer flexible and scalable solutions for literary analysis, digital humanities, and intelligent reader guidance applications.

4.2. Data Preprocessing

For NLP and ML models to function effectively, the text data must be structurally and semantically preprocessed into a format suitable for analysis. This transformation from raw to clean data significantly affects the model's learning capacity. In this study, most of the literary works were obtained in PDF format and were first converted into plain text (.txt) files. Subsequently, meaningless characters, punctuation, and structural inconsistencies were removed, and the texts were segmented into sentences and words.

The preprocessing pipeline consists of three main stages: formatting and labeling the text files, cleaning and normalization, and converting the data into a format compatible with the Doc2Vec algorithm using the TaggedDocument structure. These steps are described in detail below.

4.2.1. Text Formatting and File Naming

The dataset comprises a digital collection of Turkish literary works by various authors. Since many books were originally in PDF format, they were converted to .txt using either manual extraction or Python libraries such as pdfminer or PyMuPDF. This conversion ensured line-based access to the content, enabling further NLP processing.

After the conversion, each book was renamed according to a systematic file-naming convention designed to provide meaningful labels for the Doc2Vec training process. The full filename initially contained the author's full name and the title of the book, such as Sabahattin_Ali_Kurk_Mantolu_Madonna.txt. This was shortened to a compact tag like SA8, where SA refers to the author's initials and 8 represents the book's position within the author's collection. If the same author had multiple books, they were labeled sequentially as SA1, SA2, SA3, and so on.

4.2.2. Text Cleaning and Preprocessing Steps

Each .txt file underwent a structured and rigorous preprocessing pipeline. First, all texts were normalized using Unicode NFKD (Normalization Form Compatibility Decomposition), which decomposes compound characters and ensures standardization for downstream analysis. After normalization, all punctuation marks, typographic symbols, and special characters were removed. These included not only standard ASCII punctuation but also symbols commonly found in literary writing, such as quotation marks, angled brackets, and dashes. Non-printable control characters in the ASCII range 0-31 and 127-159 were also removed.

To preserve syntactic integrity, sentence segmentation was performed using period, exclamation mark, question mark, semicolon, and newline characters. Each sentence was stripped of extra whitespace, converted to lowercase, and prepared for analysis. While number removal was defined as an optional feature in the pipeline, numeric characters were retained in this study.

Figure 6 shows a raw excerpt from a Turkish literary text before cleaning.

```
Kapı açıldı. Elimdekini çabucak masaya bırakmak istedim, fakat geç
kalmıştım. Macar şirketinden gelen mektubun tercümeleriyle bana doğru
yaklaşan Raif Efendi'ye özür diler gibi:
"Çok güzel bir resim..." dedim.
Onun şaşıracağını, sırrını ele vereceğimden korkacağını sanmıştım. Hiç
de böyle olmadı. Her zamanki yabancı ve dalgın gülüşüyle kâğıdı elimden
alarak:
"Senelerce evvel, bir müddet resimle meşgul olmuştum!.." dedi. "Ara
sıra, el alışkanlığıyla bir şeyler karalıyorum... Görüyorsunuz ya, manasız
şeyler... Can sıkıntısı işte..."
Resmi avcunun içinde buruşturarak kâğıt sepetine attı.
"Daktilo hanımlar pek acele yazdılar!" diye mırıldandı. "Herhalde
yanlışlar vardır, fakat okumaya kalksam Hamdi Bey'i daha çok
kızdıracağım... Hakkı da var... Götürüp vereyim bari..."
Tekrar dışarı çıktı. Gözlerimle kendisini takip ettim. "Hakkı da var,
hakkı da var!" diye söyleniyordum.
```

Figure 6. Sample raw text (before cleaning)

An approximate English translation of the Turkish literary excerpt shown in Figure 6, presented for illustrative purposes in the preprocessing section:

"The door opened. I tried to quickly put down what I had in my hand on the table, but I was late. Approaching me with the translated letter from the Hungarian company, Raif Efendi apologized:
'That's a very beautiful picture...' I said.
I thought he would be surprised, maybe even scared that I had discovered his secret. But he wasn't. With his usual distant and timid smile, he gently took the paper from my hand:
'Years ago, I was occupied with painting for a while...' he said. 'From time to time, I sketch things out of habit... As you can see, nonsense... Just boredom...'
He crumpled the picture in his palm and threw it into the wastebasket.
'The typists wrote it in such a hurry!' he mumbled. 'There are probably mistakes, but if I read it out loud, I'll upset Mr. Hamdi even more... He's right... I'll just deliver it.'
He went back outside. I followed him with my eyes. 'He's right, he really is right!' I kept saying to myself."

Figure 7 shows the same excerpt after being split into individual sentences.

```
['kapı açıldı',
 'elimdekini çabucak masaya bırakmak istedim fakat geç kalmıştım',
 'macar şirketinden gelen mektubun tercümeleriyle bana doğru yaklaşan raif efendiye özür diler gibi çok güzel bir resim',
 'dedim',
 'onun şaşıracağını sırrını ele vereceğimden korkacağını sanmıştım',
 'hiç de böyle olmadı',
 'her zamanki yabancı ve dalgın gülüşüyle kâğıdı elimden alarak senelerce evvel bir müddet resimle meşgul olmuştum',
 'dedi',
 'ara sıra el alışkanlığıyla bir şeyler karalıyorum',
 'görüyorsunuz ya manasız şeyler',
 'can sıkıntısı işte',
 'resmi avcunun içinde buruşturarak kâğıt sepetine attı',
 'daktilo hanımlar pek acele yazdılar',
 'diye mırıldandı',
 'herhalde yanlışlar vardır fakat okumaya kalksam hamdi beyi daha çok kızdıracağım',
 'hakkı da var',
 'götürüp vereyim bari',
 'tekrar dışarı çıktı',
 'gözlerimle kendisini takip ettim',
 'hakkı da var hakkı da var',
 'diye söyleniyordum']
```

Figure 7. Sample text segmented into clean sentences

These sentences were tokenized into individual words and structured into a format suitable for the Doc2Vec algorithm using the TaggedDocument structure. Each document included the token list and the document's assigned label.

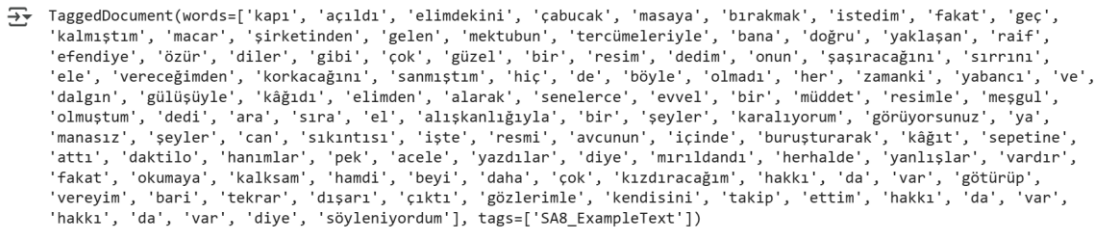

Figure 8. Sample TaggedDocument structure for Doc2Vec input

The above three figures (6, 7, and 8) illustrate how raw literary data is systematically converted into semantically and structurally coherent vectors ready for modeling. This cleaned corpus forms the basis of the author and genre classification tasks introduced in later sections of the thesis.

4.2.3. Word Cloud Visualization

To visually explore the most frequently used words and semantic themes across different authors, word clouds were generated during the data preparation phase. These visualizations were created using Python's wordcloud and matplotlib libraries. Although they were not directly used as input features for modeling, they served as an effective exploratory tool to verify the lexical richness and authorial style of each book. Word frequency patterns in these clouds provided early confirmation that the preprocessing pipeline successfully preserved stylistic integrity.

Figure 9 presents a word cloud generated from the cleaned texts of Sabahattin Ali.

Sabahattin Ali
onun
başka
tekrar
buyük
Yalnız
Maria
kadın
Efendi
Raif
yoktu
dedim
birdenbire
beraber
fazla
Demek
olduğu
kere
evvel
küçük
insan
doğru
ancak
müdet
aynı
hâlâ
karşı
gözlerini
ilk
gün
yerde
hemen
sadece
asıl
akşamdışarı
ara sıra
manasız
müthiş
kendisine
uzun
hakikaten
herhangi
kendimi
türlü
üzerinde
adeta
etmek
Halbuki
iyi
daima
dünyada
cevap
merak
sefer
yakın
Bugün
fark
güzel
oldukça
gibiydi
Maria Puder
Kürk Mantolu
sırada
Nihayet
başladı
devam
hayret
gittim
gördüm
kendime
başımı
duran
eski
Hamdi
sesle
ediyordum
Içimde
bende
gece
Raif
Berlin
ettim
yeregözleri
kadının
önüne
anda
uzak
ağır
garip
yüzüme
kız
gitmek
Yüzünde
nefes
Hasta
galiba
imkân
ettiğim
görünce
evine
olduğunu
buraya
hakkında
kuvvetli
lazım
evde
resim
siyah
sahiden
yer
beyaz
çocuk
değildi
mümkün
Dün akşam
önünde
düşündüm
derin
Ankara
olduğum
anlatım
arasında
insanın
muhakkak
Şimdiye
yanında
yanlış
yaşıyor
yaşıyordu
yaşıyorum
yaşıyorsun
yaşıyorlar
yaşıyoruz
yaşıyorsunuz
yaşıyorlardı
yaşıyordum
yaşıyordun
yaşıyordu
yaşıyorduk
yaşıyorduk
yaşıyorlardı
yaşıyordu

4.3. Embedding Algorithm

To mathematically represent the contents of books, this study employed the Doc2Vec algorithm. This model encodes both the structural and semantic features of texts into a dense vector format. As a result, each book can be used as a fixed-length numerical vector in classification, similarity analysis, and recommendation systems.

Doc2Vec extends the word-level embedding concept of Word2Vec to the document level. During model training, each book was provided to the model as a TaggedDocument object. This structure represents the book text as a list of words and a short, unique tag.

In this study, the Distributed Memory (DM) architecture was chosen, and the model was configured to generate 200-dimensional vectors for each book.

The TaggedDocument structure illustrated in Figure 8 was used directly as input to the Doc2Vec model in this phase. With this structure, each book was introduced to the model through its words and associated tag, enabling the model to learn fixed-length vectors that capture the semantic content of the text.

```
array([-0.01134722, -0.01966768, -0.02508323,  0.02726937,  0.01131074, -0.00463964, -0.01640656, -0.00170386, -0.02379159,  0.00499439,
        0.00522955,  0.00802644, -0.00717052, -0.00246213, -0.0149162 , -0.01951372,  0.00736517,  0.02215694, -0.02191249, -0.0141915 ,
       -0.01030718, -0.00109508, -0.0171658 ,  0.00426208,  0.0144895 , -0.02120543, -0.01880632, -0.03300203,  0.01135083, -0.01464216,
        0.02049132,  0.01981452, -0.01157505, -0.00389737, -0.0035953 ,  0.01047694,  0.00017214, -0.02829794, -0.01055299,  0.00070422,
       -0.00854145, -0.01918044,  0.01029937, -0.02064342,  0.01483508, -0.00760457,  0.00411719, -0.01182542,  0.01334479, -0.01822407,
       -0.00024273, -0.00242712, -0.01440999, -0.02039351, -0.00346705,  0.01619176, -0.00468175,  0.00336319, -0.02144544,  0.02172417,
       -0.00075561,  0.02167855,  0.01275052, -0.01474858, -0.00798785, -0.01102783,  0.02015183,  0.01860578, -0.01083511,  0.00064027,
       -0.02278123, -0.00166082, -0.01822779, -0.02149769, -0.01067518,  0.0059016 , -0.00949875, -0.02389922, -0.0036248 ,  0.00093972,
        0.02058857,  0.0094809 , -0.03005657,  0.01158558,  0.01369817,  0.00332204,  0.00941255, -0.00944889,  0.01627658, -0.00093004,
       -0.01055514,  0.02140762, -0.01990856, -0.01073266,  0.00350893,  0.03073958,  0.01131682, -0.0064946 ,  0.02387236,  0.00721066,
       -0.01751534,  0.00022271, -0.0104536 ,  0.00527565, -0.0201483 , -0.01097136, -0.0021115 ,  0.02188112,  0.0100689 ,  0.00508299,
       -0.00114517, -0.01703877, -0.00586062, -0.01581539,  0.021768  , -0.02445672, -0.01993903, -0.0149473 ,  0.00823292,  0.00942831,
        0.01709869,  0.02165565, -0.01090069,  0.01514396, -0.03079527,  0.00312874, -0.01559407, -0.00122324, -0.00093284, -0.02106868,
        0.01070055,  0.00456247, -0.00258408,  0.01258887, -0.01550253,  0.00541683,  0.00125696,  0.00586886, -0.02389731, -0.02176142,
        0.02243926, -0.00864758,  0.00647693, -0.01886821, -0.00232696,  0.0036141 ,  0.00620792,  0.02427666,  0.00236984, -0.00909234,
       -0.00752467,  0.02164924,  0.01987115, -0.01645418, -0.01677034,  0.02182245,  0.00301962,  0.02310649, -0.01317068, -0.00389333,
        0.02383156,  0.01706563,  0.02347864, -0.00980388,  0.00662053, -0.01344243, -0.01276521, -0.01273819, -0.02364497, -0.004496  ,
        0.00796058, -0.0021478 ,  0.0082281 ,  0.00648794,  0.02075791,  0.00956672,  0.01588753, -0.00442379, -0.01654622,  0.00381759,
        0.01595   ,  0.01401015, -0.02441265, -0.00327336, -0.01706773,  0.02235482, -0.00269278,  0.00828307,  0.02842541,  0.00361814,
       -0.01469724,  0.01895702,  0.00299676,  0.01905979, -0.00409786,  0.01173148,  0.0160422 , -0.01261286,  0.00284916,  0.00465988],
      dtype=float32)
```

Figure 10. 200-dimensional vector output for the book SA8_ExampleText

This vector provides a compact representation of the entire content of the book. While learning this vector, Doc2Vec utilized both the contextual information of the words and the overall document identity. The resulting vector will be used in the subsequent stages of the study for tasks such as author classification and genre prediction.

4.4. Model Training and Classification

The system developed in this study aims to fulfill two main tasks: automatically classifying the genre of literary texts (Genre-Finder) and identifying which well-known author a new text stylistically resembles (Author-Matcher). For both tasks, classification was performed using pre-embedded text representations.

4.4.1. Centroid-Based Classification via Vector Similarity

In addition to supervised learning techniques, this study also implements an unsupervised classification approach based on cluster centroids and vector similarity. This method is applied to both the genre prediction (Genre-Finder) and author similarity (Author-Matcher) tasks.

In this approach, document vectors obtained using the Doc2Vec algorithm are projected into a high-dimensional vector space, where each genre or author is treated as a cluster. For every genre or author represented in the training set, a centroid vector is calculated by averaging the embedding vectors of all books associated with that class. The centroid vector $Centroid_i$ for each class i is computed by averaging the embedding vectors of all books belonging to that class, as defined in Eq. (13):

$$Centroid_i = \frac{1}{L_i}\sum_{j=1}^{L_i} V_j\,, \quad \forall i = 1, \dots, M \tag{13}$$

Here, L_i denotes the number of books in the ith class (either genre or author), V_j represents the embedding vector of the jth book in that class, and M refers to the total number of distinct genres or authors. From a mathematical perspective, cosine similarity is computed as the cosine of the angle formed between two vectors, as expressed in Eq. (14), based on the general formulation previously introduced in Section 3.6 (see Eq. (4)).

$$centroid - based\ cosine\ similarity(V, Centroid_i) = \frac{\langle V, Centroid_i \rangle}{\|V\| \cdot \|Centroid_i\|}, \forall i = 1, \cdots, \tag{14}$$

In this thesis, this unsupervised approach is referred to as Centroid-Based Cosine Similarity (also sometimes referred to as clustering by other researchers), to emphasize its reliance on comparing vectors to class centroids. Each test book is assigned to the class whose centroid has the highest cosine similarity with its own embedding vector. This method enables classification based purely on spatial proximity in the vector space, without the need for any labeled training process.

The Centroid-Based Cosine Similarity classification approach was independently applied to both the Genre-Finder and Author-Matcher systems. The performance results were then compared against those obtained through supervised machine learning models.

4.4.2. Machine Learning-Based Classification Approach

In this study, the embedding vectors derived from the Doc2Vec algorithm were used as input features for various machine learning models, while the corresponding genre or author information of each book was assigned as the target label. This framing allowed the classification task to be approached as a supervised learning problem. The same method was consistently applied for both the Genre-Finder and Author-Matcher models.

The prepared training datasets were used to train five distinct machine learning algorithms, each adapted to a softmax-based multiclass classification framework. Among these algorithms, the Linear Support Vector Classifier (LinearSVC) was employed to find an optimal hyperplane that separates classes in high-dimensional vector space. Its linear nature made it particularly suitable for dense textual embeddings such as those produced by Doc2Vec.

Logistic Regression was used to model the probability distribution of class membership across multiple classes. Extended with the softmax function, this algorithm offered advantages in terms of fast convergence and interpretability, which are valuable properties in text classification contexts.

Random Forest, an ensemble learning method based on decision trees, was included to leverage its ability to generalize over complex and non-linear data structures. Each

decision tree within the ensemble was trained on randomly selected subsets of the data and features, with the final prediction determined by majority voting. This characteristic enabled the model to accommodate the nuanced patterns often observed in textual embeddings.

The Gaussian Naive Bayes model was incorporated as a probabilistic baseline. It operates under the assumption of feature independence and normal distribution of the input values. Although it is a relatively simple model, its low computational cost and practical performance in baseline comparisons made it suitable for inclusion.

The K-Nearest Neighbors (KNN) classifier, configured with k=3, assigned labels to new instances based on the majority class among their nearest neighbors. Given the semantic richness of the Doc2Vec vectors, it was hypothesized that similar books would be mapped to proximate locations in vector space, making KNN a viable approach for capturing stylistic and thematic proximity. Furthermore, its structure allowed for visual interpretability of clustering behavior.

Each algorithm was trained independently for both genre and author classification tasks, and their performance was evaluated using standard metrics such as accuracy, F_1-score, and error rate. These evaluations enabled a comparative analysis between traditional similarity-based classification and machine learning-based prediction.

After the training phase, the embedding vectors of books in the test set were fed into the trained machine learning models. The models then produced predictions for the most probable genre or author, thereby assigning each book to the group to which it bore the strongest semantic or stylistic resemblance.

Overall, this machine learning-based classification framework aimed not only to capture vector-level similarities but also to exploit the statistical separability between classes. As a result, the approach contributed to enhancing the predictive power and robustness of the proposed recommendation system.

4.5. Performance Evaluation

To evaluate the effectiveness of the classification models developed in this study, several performance metrics were utilized, including accuracy, precision, recall, F_1-

score, and the confusion matrix. Given the multi-class nature of the classification tasks, macro-averaged versions of precision, recall, and F_1-score were employed to ensure a balanced assessment across all classes, regardless of class size. These metrics enable a comprehensive evaluation of model performance, particularly in scenarios with class imbalance. A detailed explanation of each metric, along with their mathematical formulations and justification for macro-averaging, is provided in Section 3.8 and its corresponding subsections.

4.6. Visualization: Analysis of Doc2Vec Embeddings

In order to evaluate the semantic integrity and structural organization of high-dimensional document vectors produced by the Doc2Vec model, this research utilizes two core visualization methods: heatmaps and t-distributed Stochastic Neighbor Embedding (t-SNE). These visualization strategies offer a practical view into how effectively the model has internalized both thematic and stylistic patterns.

4.6.1. Cosine Similarity Analysis via Heatmap

To visualize the internal structure of the learned vectors, cosine similarity scores were calculated between all book embeddings in the training dataset. These scores were organized into a square matrix, forming a heatmap. In this matrix, the diagonal entries (with values always equal to 1.0) reflect the book's similarity to itself, while the values off the diagonal indicate how semantically related different books are to one another. A darker shade within the heatmap corresponds to a higher degree of similarity, signaling greater semantic alignment.

This visualization exclusively targets the training data to provide a clearer picture of how well the vector space has been organized during the learning process. The test data was intentionally omitted to avoid any inference-related biases.

Upon examining the heatmap, it became clear that books written by the same author or belonging to the same literary genre formed tight clusters. This observation indicates that the model successfully captures both thematic coherence and stylistic

similarity, effectively grouping texts that share common narrative or linguistic characteristics.

4.6.2. Two-Dimensional Projection via t-SNE

To visualize the structure of the high-dimensional book embeddings generated by Doc2Vec, this study employed the t-SNE algorithm to reduce these vectors, originally in 200 dimensions-to a two-dimensional representation. This dimensionality reduction technique is especially useful for detecting and interpreting clustering behavior among data points. The t-SNE method retains the local structure of the vector space, ensuring that semantically related books appear close together, while unrelated texts are projected farther apart.

Two separate t-SNE visualizations were produced. In the first, book embeddings were colored based on their corresponding authors. It was found that books written by the same author tend to gather around a central point, indicating stylistic consistency. The second visualization grouped the books by their genres, revealing clear clusters for genres such as Detective Novel and Science Fiction, where thematic boundaries are especially well-defined.

Instead of relying on symbols, this study distinguished book groups through color coding: training data, test data, and the calculated central positions (centroids) for authors or genres were each assigned a distinct color. This approach maintained visual clarity without the need for shape-based annotations.

It should be emphasized that compressing high-dimensional vectors to two dimensions through t-SNE involves an inevitable loss of information. Therefore, these projections should not be interpreted as exact spatial representations, but rather as intuitive summaries that help uncover how the model organizes the semantic space.

4.7. Train-Test Splitting

To rigorously assess the accuracy and generalization capability of the classification models developed in this study, a two-phase evaluation strategy was implemented. The

dataset was divided to allow for both in-sample tests involving known authors and out-of-sample tests involving entirely unseen authors.

4.7.1. In-Sample Evaluation: Construction of the Training and Test Sets

Before the Doc2Vec model was trained, the dataset was divided into two parts: one for training and the other for testing. This division was performed separately for the genre classification and author attribution tasks. Unlike conventional random splitting, care was taken to ensure that every author appeared in both the training and testing sets. This was accomplished using a custom Python function named split_books().

For each author, a single book was randomly selected to form the test set, while the remaining works were allocated to the training set. This ensured consistent author representation across both subsets. When the test set size was insufficient (e.g., less than 20% of the dataset), extra books were randomly sampled to increase its volume.

This strategy made it possible to assess the model's ability to generalize to unseen works by known authors and was applied to both the Genre-Finder and Author-Matcher systems.

4.7.2. Out-of-Sample Evaluation: Genre Classification for Unseen Authors

To simulate the recommender system's actual use case, the second phase involved a separate test set comprising books by authors not present in the training set. Although the authors were new, the genre labels of these books were already known to the model.

The objective at this stage was to determine whether the model could correctly classify books into their appropriate genres based solely on semantic features, without prior knowledge of the author. This test demonstrated the system's capability to make content-based genre predictions even for entirely new literary works.

Through this two-phase evaluation approach, the first phase assessed the model's ability to recognize books by familiar authors, while the second phase evaluated its competence in genre classification for works by unfamiliar authors.

4.8. On the Role and Significance of Classification

The thematic classification process in this study not only served to label the training and test datasets but also provided a crucial foundation for interpreting outputs produced by visualization tools such as t-SNE. Thanks to this pre-labeling process, it was easier to analyze which literary genres or thematic categories the observed clusters in the model outputs corresponded to.

In conclusion, the categorization of books in this project was carried out in a practical and logical manner based on content similarities, utilizing web-based sources. While not aiming for academic-level accuracy, this approach offered a functional basis that supports the feasibility and interpretability of machine learning-based literary analyses.

4.9. Development Environment

All modeling, analysis, and visualization procedures developed in this thesis were implemented using the Google Colab platform. Google Colab provides a cloud-based and free development environment that operates on a Jupyter Notebook infrastructure, allowing code written in Python to be executed online.

Throughout the study, the virtual hardware utilized consisted of a 2-core Intel(R) Xeon(R) CPU @ 2.20GHz and 13 GB of RAM. All computational tasks were executed exclusively on the CPU, without requiring GPU support.

The programming language used in this project was Python, specifically version 3.11.12. Within this framework, text preprocessing, document vectorization, application of classification algorithms, and data visualization were performed using several Python libraries. The key libraries employed included: numpy (v1.23.5), pandas (v2.2.2), matplotlib (v3.10.0), seaborn (v0.13.2), scikit-learn (v1.6.1), and gensim (v4.3.1).

This combination of tools and libraries enabled an efficient, reproducible, and systematically organized coding process, ensuring that each stage of the thesis was carried out within a cohesive and methodical framework.

CHAPTER 5: RESULTS AND DISCUSSION

This chapter presents the results obtained from the genre and author classification experiments conducted using the proposed system. The analyses include both in-sample and out-of-sample evaluations to assess the system's performance in realistic scenarios. By examining the accuracy and generalizability of the models on various datasets, this section provides a comprehensive interpretation of the system's effectiveness. Key findings are supported by statistical metrics, confusion matrices, and comparative tables, highlighting the strengths and limitations of each classification approach.

5.1. Evaluation on In-Sample Authors

In this section, the classification performance of the proposed models is evaluated on books written by authors whose other works were already included in the training set. In other words, while these authors were seen during training, the test books themselves had not been encountered by the model. This setup allows for a robust assessment of how well the system generalizes to new works by familiar authors, both in terms of genre classification (Genre-Finder) and author style prediction (Author-Matcher).

Multiple classification approaches were applied to this task, including an unsupervised Centroid-Based Cosine Similarity method as well as five supervised machine learning algorithms: Logistic Regression, Linear Support Vector Classifier (LinearSVC), Random Forest, Gaussian Naive Bayes, and K-Nearest Neighbors (k=3). For each model, evaluation metrics such as accuracy, precision, recall, F_1-score, and confusion matrices were used to compare and interpret performance.

5.1.1. Dataset Overview

The dataset comprises literary texts written in or translated into Turkish and English. In order to facilitate meaningful learning and accurate classification, the books were

manually assigned to thematic genre categories such as Detective Novel, Classic Novel, Biography, Modern Fiction, Children's Literature, and others. These genre labels were determined based on content summaries, narrative structure, and publicly available bibliographic information.

This thematic classification served as a foundational layer not only for training the classification models, but also for interpreting the results during performance analysis and visualization. It helped the system distinguish semantic differences and stylistic traits across categories and authors more effectively.

5.1.1.1. English Dataset

Table 1. English Books Dataset

No	Author Book Count	Author	Book	Tag	Split	Genre
1	6	Agatha Christie	The Mysterious Affair at Styles	AC1	Train	Detective Novels
2		Agatha Christie	The Murder on the Links	AC2	Train	Detective Novels
3		Agatha Christie	Poirot Investigates	AC3	Train	Detective Novels
4		Agatha Christie	The Big Four	AC4	Test	Detective Novels
5		Agatha Christie	The Mystery of the Blue Train	AC5	Train	Detective Novels
6		Agatha Christie	Black Coffee	AC6	Train	Detective Novels

Table 1 (Continued). English Books Dataset

7	3	Andrew Lang	The Green Fairy Book	AL1	Train	Children and Young Adult
8		Andrew Lang	The Yellow Fairy Book	AL2	Test	Children and Young Adult
9		Andrew Lang	The Orange Fairy Book	AL3	Train	Children and Young Adult
10	5	Anna Katharine Green	The Leavenworth Case	AKG1	Test	Detective Novels
11		Anna Katharine Green	A Strange Disappearance	AKG2	Train	Detective Novels
12		Anna Katharine Green	The Mystery of the Hasty Arrow	AKG3	Train	Detective Novels
13		Anna Katharine Green	Initials Only	AKG4	Train	Detective Novels
14		Anna Katharine Green	The Doctor His Wife and the Clock	AKG5	Train	Detective Novels
15	8	Arthur Conan Doyle	A Study in Scarlet	ACD1	Test	Detective Novels
16		Arthur Conan Doyle	The Hound of the Baskervilles	ACD2	Train	Detective Novels

Table 1 (Continued). English Books Dataset

17		Arthur Conan Doyle	Stories	ACD3	Train	Detective Novels
18		Arthur Conan Doyle	The Sign of the Four	ACD4	Train	Detective Novels
19		Arthur Conan Doyle	The Memoirs of Sherlock Holmes	ACD5	Train	Detective Novels
20		Arthur Conan Doyle	The Return of Sherlock Holmes	ACD6	Train	Detective Novels
21		Arthur Conan Doyle	His Last Bow	ACD7	Train	Detective Novels
22		Arthur Conan Doyle	The Case Book of Sherlock Holmes	ACD8	Train	Detective Novels
23	5	Charles Dickens	Great Expectations	CD1	Train	Classic Novels
24		Charles Dickens	Oliver Twist	CD2	Train	Classic Novels
25		Charles Dickens	A Tale of Two Cities	CD3	Test	Classic Novels
26		Charles Dickens	David Copperfield	CD4	Train	Classic Novels
27		Charles Dickens	Hard Times	CD5	Train	Classic Novels
28	4	Christopher Marlowe	Doctor Faustus	CM1	Train	Drama Tiyatro
29		Christopher Marlowe	Tamburlaine the Great Part1	CM2	Test	Drama Tiyatro
30		Christopher Marlowe	Tamburlaine the Great Part2	CM3	Train	Drama Tiyatro
31		Christopher Marlowe	The Jew of Malta	CM4	Train	Drama Tiyatro
32	6	E E Doc Smith	Triplanetary	EEDS1	Train	Science Fiction

Table 1 (Continued). English Books Dataset

33		E E Doc Smith	The Skylark of Space	EEDS2	Train	Science Fiction
34		E E Doc Smith	Children of the Lens	EEDS3	Train	Science Fiction
35		E E Doc Smith	Galactic Patrol	EEDS4	Train	Science Fiction
36		E E Doc Smith	Gray Lensman	EEDS5	Test	Science Fiction
37		E E Doc Smith	Second Stage Lensmen	EEDS6	Train	Science Fiction
38	6	Edgar Allan Poe	Allan Poe The Raven	AP1	Train	Poetry
39		Edgar Allan Poe	The Narrative of Arthur Gordon Pym of Nantucket	AP2	Train	Detective Novels
40		Edgar Allan Poe	The Murders in the Rue Morgue	AP3	Test	Detective Novels
41		Edgar Allan Poe	The Mystery of Marie Roget	AP4	Train	Detective Novels
42		Edgar Allan Poe	The Purloined Letter	AP5	Train	Detective Novels
43		Edgar Allan Poe	Selections From Poe	AP6	Train	Detective Novels
44	4	Edgar Rice Burroughs	A Princess of Mars	ERB1	Train	Science Fiction
45		Edgar Rice Burroughs	The Gods of Mars	ERB2	Train	Science Fiction
46		Edgar Rice Burroughs	The Warlord of Mars	ERB3	Train	Science Fiction
47		Edgar Rice Burroughs	Tarzan of the Apes	ERB4	Test	Science Fiction

Table 1 (Continued). English Books Dataset

48	3	Emile Gaboriau	The Lerouge Case	EG1	Train	Detective Novels
49		Emile Gaboriau	File No 113	EG2	Train	Detective Novels
50		Emile Gaboriau	Monsieur Lecoq	EG3	Test	Detective Novels
51	3	Emily Dickinson	Poems Series 1	ED1	Train	Poetry
52		Emily Dickinson	Poems Series 2	ED2	Test	Poetry
53		Emily Dickinson	Poems Series 3	ED3	Train	Poetry
54	5	Fyodor Dostoevsky	Crime and Punishment	FD1	Test	Classic Novels
55		Fyodor Dostoevsky	The Brothers Karamazov	FD2	Train	Classic Novels
56		Fyodor Dostoevsky	The Gambler	FD3	Train	Classic Novels
57		Fyodor Dostoevsky	The Idiot	FD4	Train	Classic Novels
58		Fyodor Dostoevsky	White Nights and Other Stories	FD5	Train	Classic Novels
59	3	George Bernard Shaw	Pygmalion	GBS1	Train	Drama Tiyatro
60		George Bernard Shaw	Arms and the Man	GBS2	Test	Drama Tiyatro
61		George Bernard Shaw	Man and Superman	GBS3	Train	Drama Tiyatro

Table 1 (Continued). English Books Dataset

62	5	H G Wells	The War That Will End War	HGW1	Train	Science Fiction
63		H G Wells	The Time Machine	HGW2	Test	Science Fiction
64		H G Wells	The Invisible Man	HGW3	Train	Science Fiction
65		H G Wells	The Island of Doctor Moreau	HGW4	Train	Science Fiction
66		H G Wells	The First Men in the Moon	HGW5	Train	Science Fiction
67	5	Henrik Ibsen	A Dolls House	HI1	Test	Drama Tiyatro
68		Henrik Ibsen	Ghosts	HI2	Train	Drama Tiyatro
69		Henrik Ibsen	Hedda Gabler	HI3	Train	Drama Tiyatro
70		Henrik Ibsen	The Wild Duck	HI4	Train	Drama Tiyatro
71		Henrik Ibsen	An Enemy of the People	HI5	Train	Drama Tiyatro
72	5	Jane Austen	Sense and Sensibility	JA1	Train	Classic Novels
73		Jane Austen	Pride and Prejudice	JA2	Train	Classic Novels
74		Jane Austen	Emma	JA3	Train	Classic Novels
75		Jane Austen	Mansfield Park	JA4	Train	Classic Novels
76		Jane Austen	Northanger Abbey	JA5	Test	Classic Novels

Table 1 (Continued). English Books Dataset

77	4	Jules Verne	Twenty Thousand Leagues Under the Sea	JV1	Train	Science Fiction
78		Jules Verne	Around The World in Eighty Days	JV2	Train	Science Fiction
79		Jules Verne	From the Earth to the Moon	JV3	Test	Science Fiction
80		Jules Verne	The Mysterious Island	JV4	Train	Science Fiction
81	3	L Frank Baum	The Marvelous Land of Oz	LFB1	Train	Children and Young Adult
82		L Frank Baum	Ozma of Oz	LFB2	Train	Children and Young Adult
83		L Frank Baum	The Wonderful Wizard of Oz	LFB3	Test	Children and Young Adult
84	5	Lewis Carroll	Alice's Adventures in Wonderland	LC1	Train	Children and Young Adult
85		Lewis Carroll	Through the Looking Glass	LC2	Train	Children and Young Adult
86		Lewis Carroll	Phantasmagoria and Other Poems	LC3	Train	Poetry

Table 1 (Continued). English Books Dataset

87		Lewis Carroll	Sylvie and Bruno	LC4	Train	Children and Young Adult
88		Lewis Carroll	The Hunting of the Snark	LC5	Test	Poetry
89	3	Louisa May Alcott	Little Women	LMA1	Test	Children and Young Adult
90		Louisa May Alcott	Jos Boys	LMA2	Train	Children and Young Adult
91		Louisa May Alcott	Little Men	LMA3	Train	Children and Young Adult
92	9	Mary Shelley	Frankenstein	MS1	Train	Classic Novels
93		Mary Shelley	The Last Man	MS2	Train	Science Fiction
94		Mary Shelley	Lodore 1	MS3	Train	Classic Novels
95		Mary Shelley	Lodore 2	MS4	Train	Classic Novels
96		Mary Shelley	Lodore 3	MS5	Train	Classic Novels
97		Mary Shelley	The Fortunes of Perkin Warbeck	MS6	Train	Classic Novels
98		Mary Shelley	Valperga 1	MS7	Train	Classic Novels

Table 1 (Continued). English Books Dataset

99		Mary Shelley	Valperga 2	MS8	Train	Classic Novels
100		Mary Shelley	Valperga 3	MS9	Test	Classic Novels
101	7	Oscar Wilde	The Picture of Dorian Gray	OW1	Train	Classic Novels
102		Oscar Wilde	The Happy Prince and Other Tales	OW2	Train	Classic Novels
103		Oscar Wilde	A Woman of No Importance	OW3	Test	Drama Tiyatro
104		Oscar Wilde	Lady Windermeres Fan	OW4	Train	Drama Tiyatro
105		Oscar Wilde	The Importance of Being Earnest	OW5	Train	Drama Tiyatro
106		Oscar Wilde	Lady Windermeres Fan	OW6	Train	Drama Tiyatro
107		Oscar Wilde	Salome	OW7	Train	Drama Tiyatro
108	3	R Austin Freeman	The Red Thumb Mark	RAF1	Train	Detective Novels
109		R Austin Freeman	John Thorndyke Cases	RAF2	Train	Detective Novels
110		R Austin Freeman	The Eye of Osiris	RAF3	Test	Detective Novels
111	4	Robert Frost	A Boy's Will	RF1	Train	Poetry
112		Robert Frost	North of Boston	RF2	Train	Poetry
113		Robert Frost	Mountain Interval	RF3	Test	Poetry
114		Robert Frost	New Hampshire	RF4	Train	Poetry
115	3	Robert Louis Stevenson	The Rajahs Diamond	RLS1	Train	Classic Novels

Table 1 (Continued). English Books Dataset

116		Robert Louis Stevenson	Treasure Island	RLS2	Test	Classic Novels
117		Robert Louis Stevenson	Kidnapped	RLS3	Train	Classic Novels
118	7	Rudyard Kipling	Just So Stories	RK1	Train	Children and Young Adult
119		Rudyard Kipling	The Jungle Book	RK2	Train	Children and Young Adult
120		Rudyard Kipling	Captains Courageous	RK3	Train	Children and Young Adult
121		Rudyard Kipling	Kim	RK4	Test	Children and Young Adult
122		Rudyard Kipling	Land and Sea Tales for Boys and Girls	RK5	Train	Children and Young Adult
123		Rudyard Kipling	Stalky and Co	RK6	Train	Children and Young Adult
124		Rudyard Kipling	The Second Jungle Book	RK7	Train	Children and Young Adult

Table 1 (Continued). English Books Dataset

125	5	Victor Hugo	Les Misérables	VH1	Train	Classic Novels
126		Victor Hugo	The Hunchback of Notre-Dame	VH2	Train	Classic Novels
127		Victor Hugo	Ninety Three	VH3	Train	Classic Novels
128		Victor Hugo	The Man Who Laughs	VH4	Train	Classic Novels
129		Victor Hugo	Toilers of the Sea	VH5	Test	Classic Novels
130	4	Wilkie Collins	The Woman in White	WC1	Train	Classic Novels
131		Wilkie Collins	The Moonstone	WC2	Test	Classic Novels
132		Wilkie Collins	Armadale	WC3	Train	Classic Novels
133		Wilkie Collins	No Name	WC4	Train	Classic Novels
134	7	William Shakespeare	Romeo and Juliet	WS1	Train	Drama Tiyatro
135		William Shakespeare	Hamlet	WS2	Train	Drama Tiyatro
136		William Shakespeare	Macbeth	WS3	Train	Drama Tiyatro
137		William Shakespeare	Othello	WS4	Train	Drama Tiyatro
138		William Shakespeare	King Lear	WS5	Train	Drama Tiyatro
139		William Shakespeare	A Midsummer Nights Dream	WS6	Test	Drama Tiyatro
140		William Shakespeare	The Tempest	WS7	Train	Drama Tiyatro

The English dataset used in this study comprises 140 literary works written by 29 different authors. These texts were classified into six major genres: Classic Novels, Detective Novels, Children and Young Adult, Drama/Theatre, Poetry, and Science Fiction. The classification was based on content similarity and literary genre characteristics. To assign each book to a genre, various online sources such as Goodreads, Wikipedia, and book retail websites were consulted. The decision was informed by thematic summaries, plot descriptions, and reader reviews. Rather than relying on strict academic or literature-based genre definitions, this approach focused on how similar works are commonly categorized in publicly accessible archives. This practical method offered a flexible yet consistent framework for labeling, which supported both the training of the classification models and the interpretability of their outputs during evaluation.

For the English dataset, 111 books were used for model training and 29 books were reserved for testing. The test set was carefully constructed to ensure that each author and each genre was represented by at least one book. As a result, 80% of the dataset was allocated to training, and the remaining 20% was used for validation.

5.1.1.2. Turkish Dataset

Table 2. Turkish Books Dataset

No	Author Book Count	Author	Book	Tag	Split	Genre
1	7	Ahmet Ümit	Aşkımız Eski Bir Roman	AU1	Train	Detective Novel
2		Ahmet Ümit	İstanbul Hatırası	AU2	Train	Detective Novel
3		Ahmet Ümit	Kavim	AU3	Train	Detective Novel
4		Ahmet Ümit	Kayıp Tanrılar Ülkesi	AU4	Train	Detective Novel

Table 2 (Continued). Turkish Books Dataset

5		Ahmet Ümit	Kırlangıç Çığlığı	AU5	Train	Detective Novel
6		Ahmet Ümit	Patasana	AU6	Train	Detective Novel
7		Ahmet Ümit	Sultanı Öldürmek	AU7	Test	Detective Novel
8	5	Ayfer Tunç	Evvel Otel Saklı	AT1	Train	Modern Novel
9		Ayfer Tunç	Kapak Kızı	AT2	Train	Modern Novel
10		Ayfer Tunç	Osman	AT3	Train	Modern Novel
11		Ayfer Tunç	Suzan Defter	AT4	Test	Modern Novel
12		Ayfer Tunç	Yeşil Peri Gecesi	AT5	Train	Modern Novel
13	4	Ayşe Kulin	Füreya	AK1	Train	Biography Memoir
14		Ayşe Kulin	Handan	AK2	Train	Modern Novel
15		Ayşe Kulin	Her Yerde Kan Var	AK3	Train	Modern Novel
16		Ayşe Kulin	Türkan	AK4	Test	Biography Memoir
17	5	Aziz Nesin	Kaç Aptalız	AN1	Test	Social Critique
18		Aziz Nesin	Ölmüş Eşek	AN2	Train	Social Critique
19		Aziz Nesin	Şimdiki Çocuklar Harika	AN3	Train	Social Critique
20		Aziz Nesin	Yaşar Ne Yaşar Ne Yaşamaz	AN4	Train	Social Critique

Table 2 (Continued). Turkish Books Dataset

21		Aziz Nesin	Zübük	AN5	Train	Social Critique
22	9	Doğan Cüceloğlu	Başarıya Götüren Aile	DC1	Train	Personal Development
23		Doğan Cüceloğlu	Evlenmeden Önce	DC2	Train	Personal Development
24		Doğan Cüceloğlu	İçimizdeki Çocuk	DC3	Train	Personal Development
25		Doğan Cüceloğlu	İletişim Donanımları	DC4	Train	Personal Development
26		Doğan Cüceloğlu	Keşkesiz Bir Yaşam İçin	DC9	Test	Personal Development
27		Doğan Cüceloğlu	Korku Kültürü	DC5	Train	Personal Development
28		Doğan Cüceloğlu	Mis Gibi Yaşamlar	DC6	Train	Personal Development
29		Doğan Cüceloğlu	Onlar Benim Kahramanım	DC7	Train	Personal Development
30		Doğan Cüceloğlu	Savaşçı	DC8	Train	Personal Development
31	6	Elif Şafak	Baba ve Piç	ES1	Train	Modern Novel
32		Elif Şafak	Havva'nın Üç Kızı	ES2	Train	Modern Novel
33		Elif Şafak	Med Cezir	ES3	Test	Modern Novel
34		Elif Şafak	Şehrin Aynaları	ES4	Train	Modern Novel
35		Elif Şafak	Siyah Süt	ES5	Train	Biography Memoir
36		Elif Şafak	Ustam ve Ben	ES6	Train	Modern Novel

Table 2 (Continued). Turkish Books Dataset

37	3	Halide Edib Adıvar	Ateşten Gömlek	HEA2	Test	Classic Novel
38		Halide Edib Adıvar	Handan	HEA1	Train	Classic Novel
39		Halide Edib Adıvar	Sinekli Bakkal	HEA3	Train	Classic Novel
40	7	Halil İnalcık	Atatürk ve Demokratik Türkiye	HI1	Train	History
41		Halil İnalcık	Devleti Aliyye Osmanlı İmparatorluğu Üzerine Araştırmalar 1	HI3	Train	History
42		Halil İnalcık	Kısa Osmanlı Tarihi	HI4	Train	History
43		Halil İnalcık	Osmanlı İmparatorluğu'nun Ekonomik ve Sosyal Tarihi 1	HI5	Test	History
44		Halil İnalcık	Osmanlı İmparatorluğu'nun Ekonomik ve Sosyal Tarihi 2	HI6	Train	History
45		Halil İnalcık	Osmanlı Tarihinde Efsaneler ve Gerçekler	HI7	Train	History
46		Halil İnalcık	Osmanlı ve Modern Türkiye	HI2	Train	History

Table 2 (Continued). Turkish Books Dataset

47	9	Hüseyin Rahmi Gürpınar	Kuyruklu Yıldız	HRG1	Train	Satire Humour
48		Hüseyin Rahmi Gürpınar	Metres	HRG2	Train	Satire Humour
49		Hüseyin Rahmi Gürpınar	Mezarından Kalkan Şehit	HRG3	Test	Satire Humour
50		Hüseyin Rahmi Gürpınar	Muhabbet Tılsımı	HRG4	Train	Satire Humour
51		Hüseyin Rahmi Gürpınar	Mürebbiye	HRG5	Train	Satire Humour
52		Hüseyin Rahmi Gürpınar	Namuslu Kokotlar	HRG6	Train	Satire Humour
53		Hüseyin Rahmi Gürpınar	Nimetşinas	HRG7	Train	Satire Humour
54		Hüseyin Rahmi Gürpınar	Ölüler Yaşıyorlar mı	HRG8	Test	Satire Humour
55		Hüseyin Rahmi Gürpınar	Ölüm Bir Kurtuluş mudur	HRG9	Train	Satire Humour
56	9	İlber Ortaylı	Avrupa ve Biz	IO1	Train	History
57		İlber Ortaylı	Batılılaşma Yolunda	IO2	Train	History

Table 2 (Continued). Turkish Books Dataset

58		İlber Ortaylı	Gazi Mustafa Kemal Atatürk	IO3	Test	History
59		İlber Ortaylı	İmparatorluğun En Uzun Yüzyılı	IO4	Train	History
60		İlber Ortaylı	Osmanlı'da Milletler ve Diplomasi	IO5	Test	History
61		İlber Ortaylı	Tarih Yazıcılık Üzerine	IO6	Train	History
62		İlber Ortaylı	Türkiye'nin Yakın Tarihi	IO7	Train	History
63		İlber Ortaylı	Türklerin Tarihi 1	IO8	Train	History
64		İlber Ortaylı	Türklerin Tarihi 2	IO9	Train	History
65	4	İskender Pala	İki Dirhem Bir Çekirdek	IP1	Train	Essay Culture
66		İskender Pala	Katre-i Matem	IP2	Train	Historical Fiction
67		İskender Pala	Müstesna Güzeller	IP3	Train	Essay Culture
68		İskender Pala	Od	IP4	Test	Historical Fiction
69	4	İsmail Gezgin	Homo Narrans	IG1	Train	Essay Culture
70		İsmail Gezgin	Masalların Şifresi	IG2	Train	Essay Culture
71		İsmail Gezgin	Ötekilerin Arkeolojisi	IG3	Train	Essay Culture
72		İsmail Gezgin	Sanatın Mitolojisi	IG4	Test	Essay Culture

Table 2 (Continued). Turkish Books Dataset

73	9	Mehmet Rauf	Bir Aşkın Tarihi	MR1	Train	Classic Novel
74		Mehmet Rauf	Böğürtlen	MR2	Train	Classic Novel
75		Mehmet Rauf	Define ve Kan Damlası	MR3	Train	Classic Novel
76		Mehmet Rauf	Ferdayı Garam	MR4	Train	Classic Novel
77		Mehmet Rauf	Genç Kız Kalbi	MR5	Train	Classic Novel
78		Mehmet Rauf	Halas	MR6	Train	Classic Novel
79		Mehmet Rauf	Hanımlar Arasında	MR7	Train	Classic Novel
80		Mehmet Rauf	Harabeler	MR8	Test	Classic Novel
81		Mehmet Rauf	Kâbus	MR9	Train	Classic Novel
82	3	Nermin Yıldırım	Misafir	NY3	Train	Modern Novel
83		Nermin Yıldırım	Unutma Beni Apartmanı	NY1	Train	Modern Novel
84		Nermin Yıldırım	Unutma Dersleri	NY2	Test	Modern Novel
85	3	Oğuz Atay	Bir Bilim Adamının Romanı	OA3	Train	Biography Memoir
86		Oğuz Atay	Korkuyu Beklerken	OA1	Test	Short Story
87		Oğuz Atay	Tutunamayanlar	OA2	Train	Modern Novel
88	4	Orhan Pamuk	Kar	OP1	Train	Modern Novel

Table 2 (Continued). Turkish Books Dataset

89		Orhan Pamuk	Kara Kitap	OP2	Test	Modern Novel
90		Orhan Pamuk	Masumiyet Müzesi	OP3	Train	Modern Novel
91		Orhan Pamuk	Sessiz Ev	OP4	Train	Modern Novel
92	9	Sabahattin Ali	Bütün Öyküleri 1	SA1	Test	Short Story
93		Sabahattin Ali	Bütün Öyküleri 2	SA2	Train	Short Story
94		Sabahattin Ali	Çakıcının İlk Kurşunu	SA3	Train	Social Critique
95		Sabahattin Ali	Canım Aliye Ruhum Filiz	SA4	Train	Biography Memoir
96		Sabahattin Ali	Değirmen	SA5	Train	Short Story
97		Sabahattin Ali	İçimizdeki Şeytan	SA6	Test	Social Critique
98		Sabahattin Ali	Kamyon	SA7	Train	Short Story
99		Sabahattin Ali	Kürk Mantolu Madonna	SA8	Train	Modern Novel
100		Sabahattin Ali	Markopaşa Yazıları ve Ötekiler	SA9	Train	Social Critique
101	4	Samed Behrengi	Bir Günlük Düş ve Gerçek	SB1	Train	Children Youth
102		Samed Behrengi	Bir Şeftali Bin Şeftali	SB2	Test	Children Youth
103		Samed Behrengi	Kel Güvercinci	SB3	Test	Children Youth
104		Samed Behrengi	Pancarcı Çocuk	SB4	Train	Children Youth

Table 2 (Continued). Turkish Books Dataset

105	3	Tarık Buğra	Firavun İmanı	TB1	Test	Classic Novel
106		Tarık Buğra	Küçük Ağa	TB2	Train	Classic Novel
107		Tarık Buğra	Yağmur Beklerken	TB3	Train	Classic Novel
108	9	Yaşar Kemal	İnce Memed 1	YK1	Train	Modern Novel
109		Yaşar Kemal	İnce Memed 2	YK2	Train	Modern Novel
110		Yaşar Kemal	İnce Memed 3	YK3	Train	Modern Novel
111		Yaşar Kemal	İnce Memed 4	YK4	Train	Modern Novel
112		Yaşar Kemal	Kanın Sesi	YK5	Test	Modern Novel
113		Yaşar Kemal	Kuşlar da Gitti	YK6	Train	Modern Novel
114		Yaşar Kemal	Sarı Sıcak	YK7	Train	Modern Novel
115		Yaşar Kemal	Sevmek Sevinmek İyi Şeyler Üstüne	YK8	Train	Modern Novel
116		Yaşar Kemal	Tek Kanatlı Kuş	YK9	Train	Modern Novel
117	9	Zülfü Livaneli	Balıkçı ve Oğlu	ZL1	Train	Modern Novel
118		Zülfü Livaneli	Bir Kedi Bir Adam Bir Ölüm	ZL2	Train	Modern Novel
119		Zülfü Livaneli	Edebiyat Mutluluktur	ZL3	Train	Essay Culture

Table 2 (Continued). Turkish Books Dataset

120		Zülfü Livaneli	Engereğin Gözü	ZL4	Train	Modern Novel
121		Zülfü Livaneli	Kaplanın Sırtında	ZL5	Test	Modern Novel
122		Zülfü Livaneli	Kardeşimin Hikâyesi	ZL6	Train	Modern Novel
123		Zülfü Livaneli	Konstantiniyye Oteli	ZL7	Train	Modern Novel
124		Zülfü Livaneli	Leylâ'nın Evi	ZL8	Train	Modern Novel
125		Zülfü Livaneli	Mutluluk	ZL9	Train	Modern Novel

The Turkish dataset consists of 125 literary works written by 21 different authors. These books were classified into twelve genres based on their content, themes, and the general literary tendencies of their authors. The genres are as follows: Classic Novel, Modern Novel, Detective Novel, Short Story, History, Social Critique, Personal Development, Children and Youth, Biography and Memoir, Historical Fiction, Essay and Culture, and Satire and Humour. Although the classification was not based directly on an academic dataset, each book was classified by carefully examining its overall content, narrative style, thematic focus, and the author's broader literary orientation. This process involved reviewing various book promotion websites, publisher descriptions, back cover texts, and user reviews. As a result, each book was assigned to a single genre deemed most appropriate. For example, Hüseyin Rahmi Gürpınar's socially critical and humorous works were placed in the Satire and Humour genre, Halil İnalcık's historical analyses in History, Ayşe Kulin's biographical narratives in Biography and Memoir, and Sabahattin Ali's short story collections in Short Story.

Out of the 125 Turkish books, 100 were used for training and 25 were allocated to the test set. The test set was constructed to ensure that at least one book from each author was included, thereby preserving both author and genre diversity. This allocation

reflects an 80/20 split between training and validation data, enabling both robust model training and meaningful evaluation.

5.2. Model Results and Analysis

This section presents the experimental results obtained from the proposed system, offering a comprehensive analysis of its performance across various evaluation setups. The subsections that follow detail the configuration of the Doc2Vec model, the visual examination of document embeddings using heatmaps and t-SNE projections, and the quantitative classification results for both genre and author prediction tasks. Additionally, out-of-sample evaluations are discussed to assess the generalizability of the system to entirely new authors. These analyses collectively provide valuable insights into the strengths and potential limitations of the developed models.

5.2.1. Doc2Vec Configuration Overview

In this study, the Doc2Vec algorithm was employed to generate dense vector representations of entire books, effectively capturing both semantic content and stylistic features. Following a series of experimental trials (specifically, ten different configuration tests conducted separately for both Turkish and English datasets), the configuration that yielded the most consistent and accurate results was selected and used across all classification tasks. The optimal settings for the Doc2Vec model included a vector size of 200, a context window size of 5, a minimum word count (min_count) of 5, 20 training epochs, and the use of the Distributed Memory (DM) architecture, indicated by dm=1.

This configuration was chosen to strike a balance between semantic richness and computational efficiency. A vector size of 200 offers sufficient dimensionality to encode complex literary features without leading to overfitting. A window size of 5 allows the model to consider a reasonable contextual span for each word, which is particularly useful in capturing stylistic and thematic patterns in prose. The use of the

DM architecture helps preserve word order and contextual dependencies, both of which are critical in modeling literary text accurately.

These parameters were applied consistently to both the Turkish and English datasets, thereby ensuring a standardized framework for implementing the Genre-Finder and Author-Matcher models. The uniformity in vector generation laid a robust foundation for the subsequent classification and visualization processes carried out in this research.

5.2.2. Heatmap Analysis

To assess the semantic and stylistic relationships between books in each dataset, heatmap visualizations of cosine similarity scores were generated. The following subsections separately analyze the English and Turkish training sets, providing insights into intra-author coherence and cross-author thematic proximity.

5.2.2.1. Heatmap Analysis - English Dataset

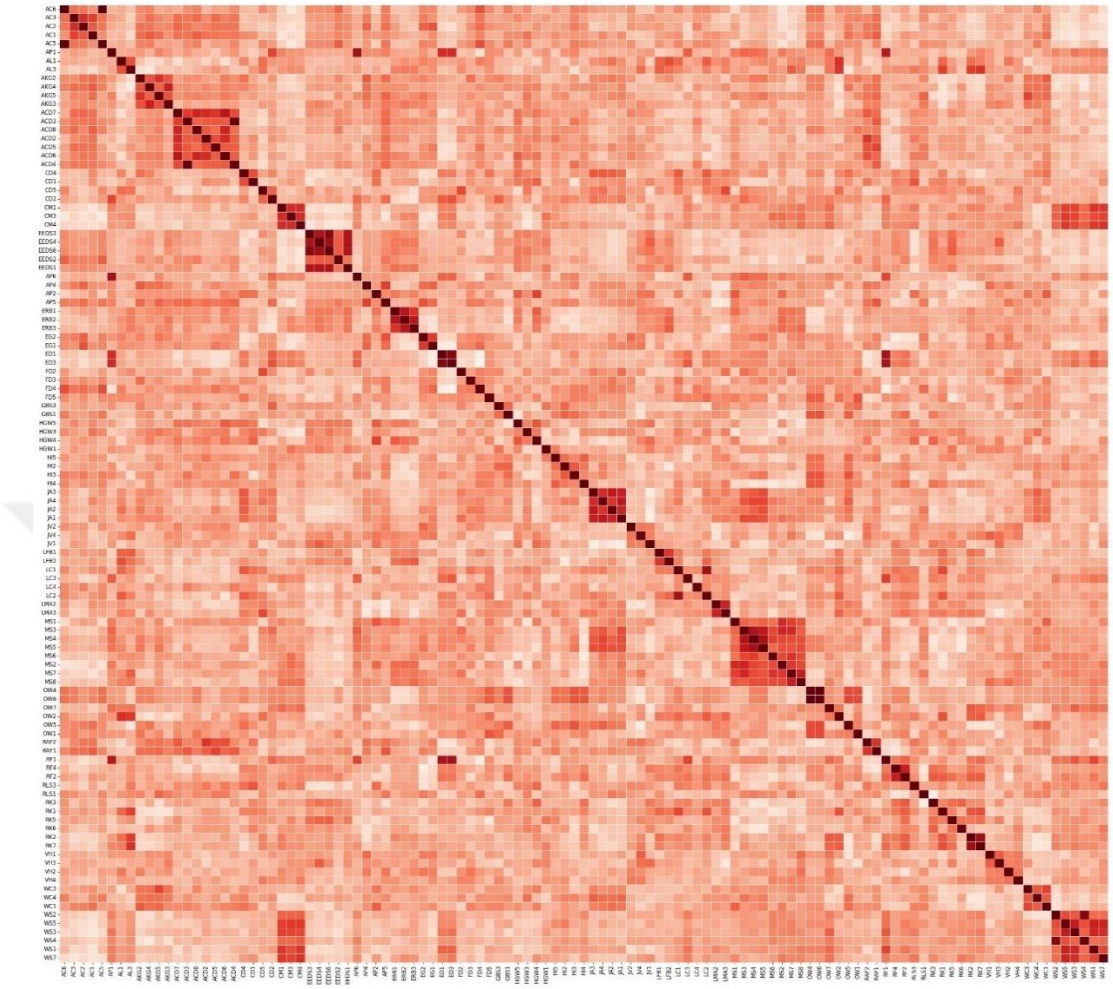

Figure 11. Cosine similarity heatmap of English training set book vectors. (This heatmap visualizes the semantic and stylistic similarities between English books in the training set. Darker regions indicate higher similarity, often observed between books by the same author.)

To visually examine the semantic and stylistic consistency of the Doc2Vec representations, a heatmap of cosine similarities between all books in the English training set was generated. Each cell in the matrix represents the cosine similarity between a pair of book vectors, with darker red colors indicating higher similarity. The diagonal line, where each book is compared to itself, consistently shows a value of 1.0, as expected.

A notable pattern in the heatmap is the presence of clearly visible darker blocks beyond the diagonal. These concentrated regions of higher similarity typically correspond to

books written by the same author. This suggests that the Doc2Vec model has effectively captured distinctive writing styles and thematic coherence across an author's works. For example, authors such as Agatha Christie or H. G. Wells show strong intra-author clustering, with their respective books forming dense, dark squares along the diagonal region.

These observations support the model’s ability to learn fine-grained stylistic and semantic features that are consistent within an author's body of work. Furthermore, the varying intensity of off-diagonal elements highlights the diversity in writing style and content across authors and genres in the English dataset.

5.2.2.2. Heatmap Analysis - Turkish Dataset

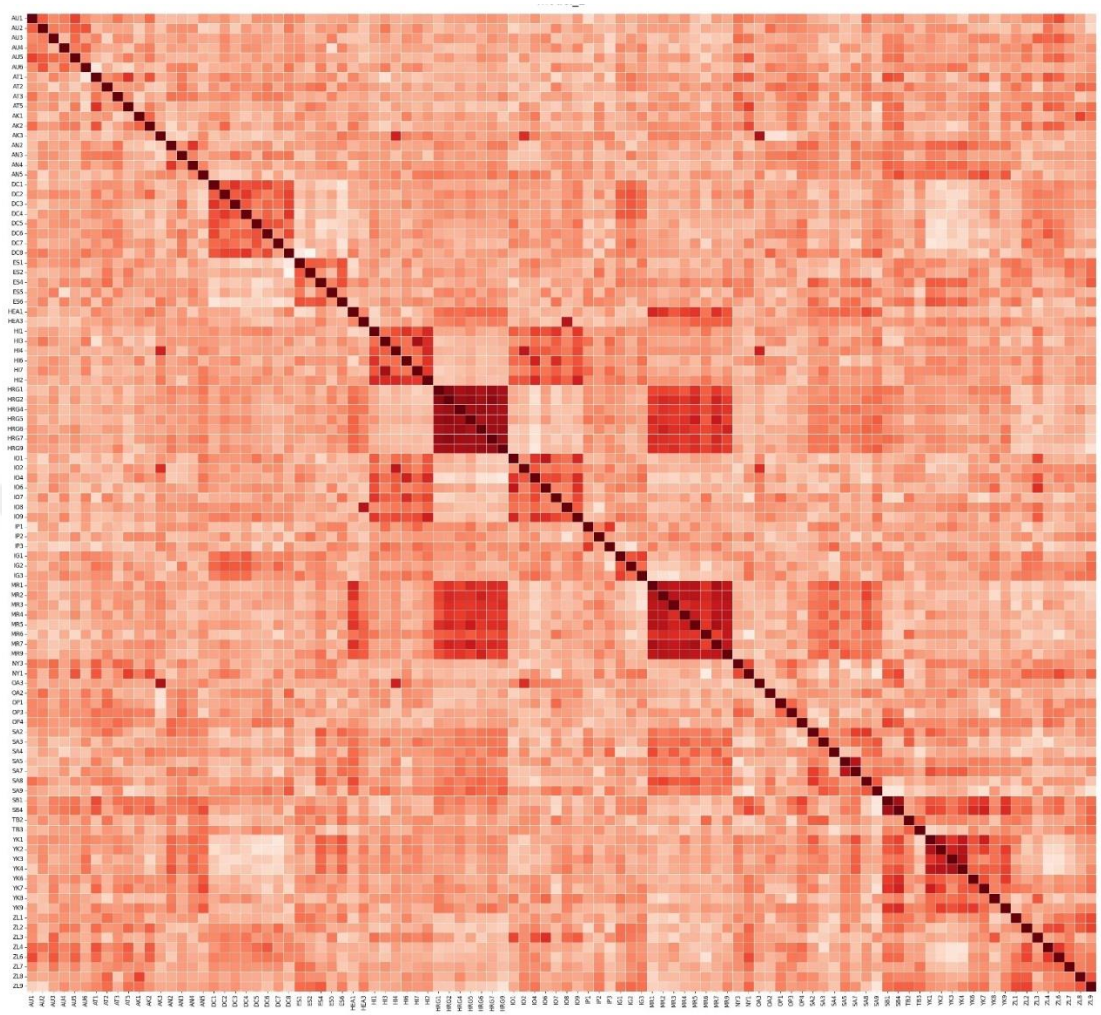

Figure 12. Cosine similarity heatmap of Turkish training set book vectors. (This heatmap displays the cosine similarity scores between Turkish books in the training set. Clusters with stronger color intensities reflect semantic or stylistic similarity, frequently aligned with author-based groupings.)

This heatmap reveals several notable semantic patterns within the Turkish dataset. Among the most prominent observations is the high degree of similarity not only within the individual works of Mehmet Rauf and Hüseyin Rahmi Gürpınar, but also between their respective corpora. This strong cross-author similarity likely stems from their shared historical context and the more elaborate, Ottoman-influenced linguistic style characteristic of late 19th and early 20th-century Turkish literature. Another significant finding is the high similarity between the works of İlber Ortaylı and Halil

İnalcık, both of whom are known for their scholarly focus on Turkish history. These results highlight the model's ability to effectively capture stylistic and thematic consistencies, especially among authors with similar eras, topics, or linguistic conventions. As expected, nearly every author shows the highest similarity within their own books, reinforcing the reliability of the document embeddings produced by the Doc2Vec algorithm.

While the Doc2Vec-based similarity matrix provides valuable insight into authorial and thematic proximity, it must be interpreted with caution. Aside from the trivial case of a book being identical to itself (similarity score of 1.0 on the diagonal), most observed inter-book similarity values lie in the moderate range of approximately 0.5 to 0.6. Although this reflects meaningful semantic overlap, it is not sufficient on its own to draw firm classification boundaries. Consequently, the inclusion of additional techniques-such as centroid-based similarity analysis and supervised machine learning models-is essential to improve classification robustness and enhance the interpretability of results.

5.2.3. Genre Visualization with t-SNE

This subsection provides a visual examination of genre-level clustering in both English and Turkish datasets using t-SNE projections. These plots illustrate how Doc2Vec embeddings capture thematic and stylistic distinctions across genres.

5.2.3.1. English Dataset - Genre t-SNE

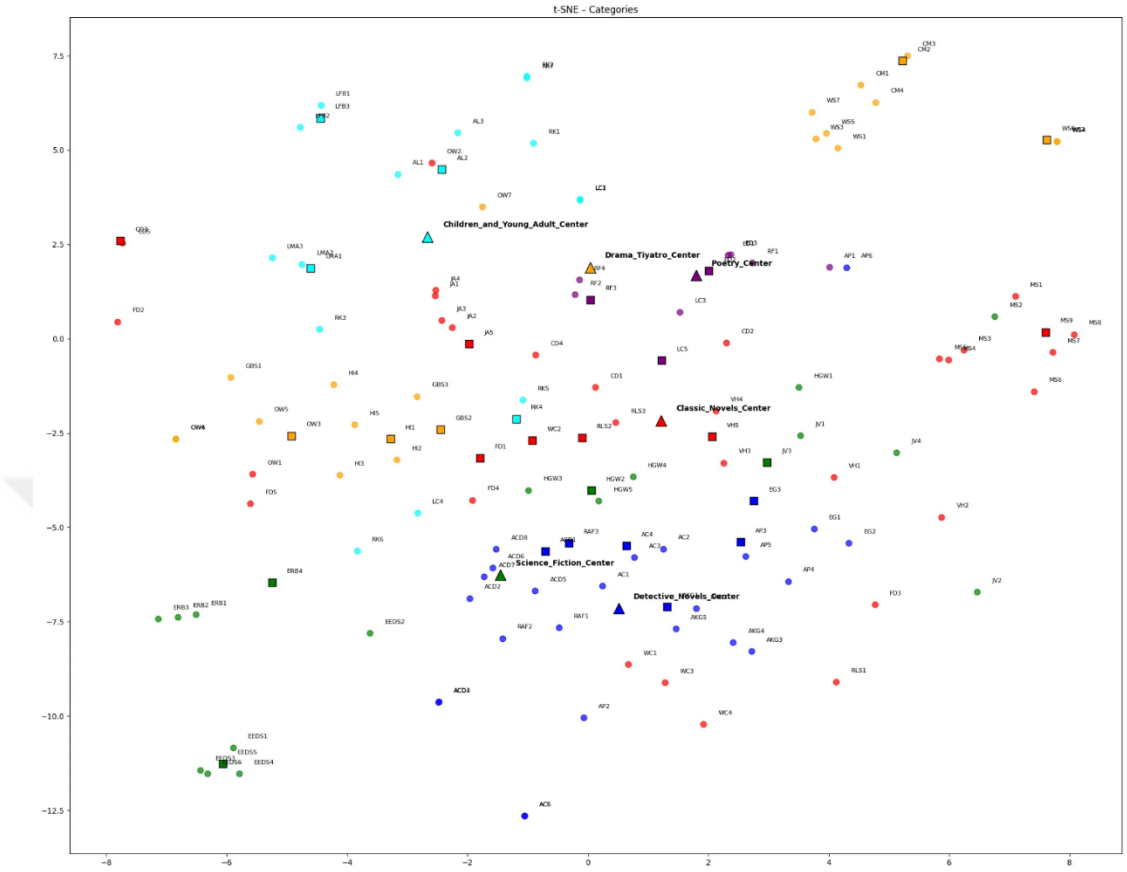

Figure 13. t-SNE visualization of English book vectors grouped by genres. Each point represents a book, and each triangle denotes the centroid of a genre.

Figure 13 presents a two-dimensional t-SNE visualization of the English training dataset, grouped by genre. Each point represents a book, with colors and shapes corresponding to its genre. The genre centroids are marked with labeled triangles. The most clearly defined clusters appear in genres with distinct narrative structures, such as Poetry and Detective Novels. Books within these genres are tightly grouped around their respective centroids, indicating that the model successfully learned their semantic and stylistic coherence.

However, since t-SNE reduces high-dimensional vectors into two dimensions, some degree of information loss is inevitable. Therefore, the resulting layout should be interpreted as an approximate visualization rather than a precise reflection of semantic distance.

The Classic Novels cluster, while more diffuse, still shows partial separation from other genres, suggesting that older literary works carry stylistic features captured by the Doc2Vec embeddings. In contrast, genres like Children and Young Adult and Drama/Tiyatro show a relatively scattered distribution, likely due to more overlapping themes or language use with other genres.

Overall, the visualization confirms that the Doc2Vec model is capable of grouping semantically and stylistically similar English books, particularly for genres with strong thematic or narrative boundaries.

5.2.3.2. Turkish Dataset - Genre t-SNE

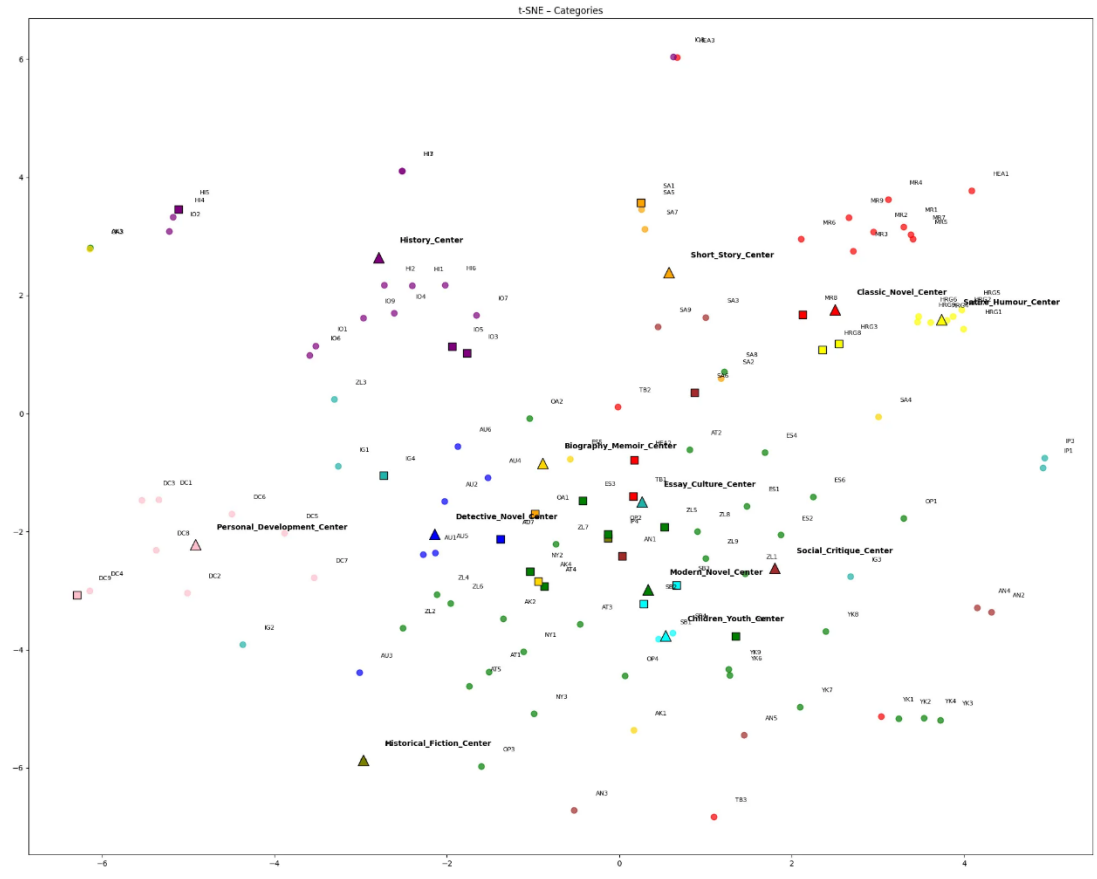

Figure 14. t-SNE visualization of Turkish book vectors grouped by genres. Each point represents a book, and each triangle denotes the centroid of a genre.

Figure 14 shows the t-SNE projection of the Turkish dataset, offering a more complex and nuanced picture due to the higher number of genres. Despite the increase in genre diversity, several well-formed clusters are clearly visible. For instance, Detective_Novel, History, and Personal_Development books are tightly grouped and centered around their respective centroids, demonstrating strong intra-genre consistency.

Notably, Satire_Humour and Short_Story genres also form visible clusters, though slightly more dispersed. This likely reflects the stylistic idiosyncrasies and short-format nature of these texts. The Modern_Novel and Social_Critique genres appear more entangled, suggesting that the stylistic boundary between these genres is less pronounced, possibly due to overlapping social themes and contemporary language usage.

The visualization highlights how Doc2Vec embeddings, combined with t-SNE, can reveal latent thematic and stylistic structures in Turkish literary texts. The overall layout suggests that while some genre boundaries are clearly encoded in the vector space, others exhibit semantic bleed-through, reinforcing the need for complementary classification methods.

5.2.4. Author Visualization with t-SNE

This subsection presents the author-level clustering results obtained from the t-SNE projections, showcasing how the Doc2Vec embeddings capture stylistic and thematic similarities among different authors in both English and Turkish datasets.

5.2.4.1. English Dataset - Author t-SNE

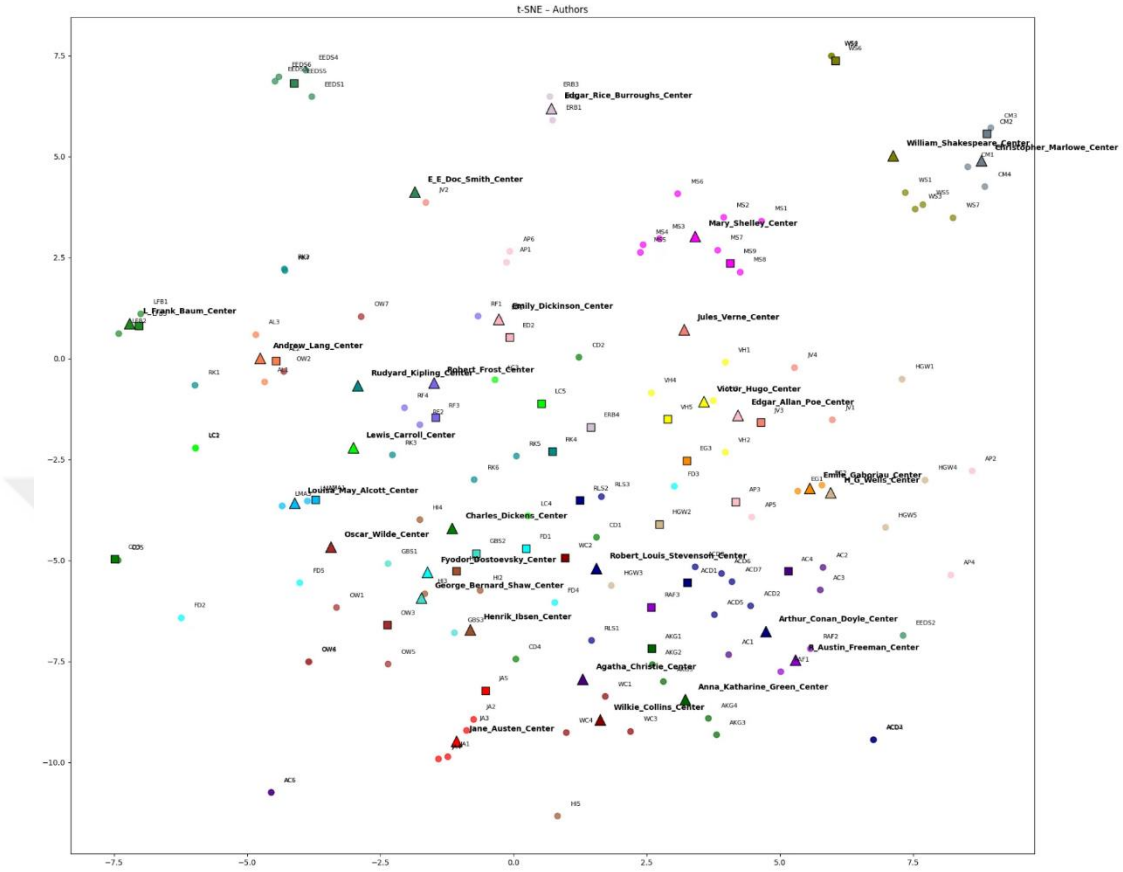

Figure 15. t-SNE visualization of English book vectors grouped by author identities. Each point represents a book, and each triangle denotes the centroid of an author.

This t-SNE plot illustrates the distribution of English-language books based on their Doc2Vec embeddings, where each color represents a different author and triangles indicate the centroid of each author's corpus. The model demonstrates a strong ability to cluster works from the same author, with clear separations visible for Agatha Christie, Oscar Wilde, and Robert Louis Stevenson. Furthermore, authors such as William Shakespeare and Victor Hugo also appear cohesively grouped, indicating the model's effectiveness in capturing shared stylistic traits and thematic consistencies. Overall, the clustering reflects meaningful semantic patterns, particularly among authors who share historical or literary traditions, such as Gothic or Romantic fiction.

5.2.4.2. Turkish Dataset - Author t-SNE

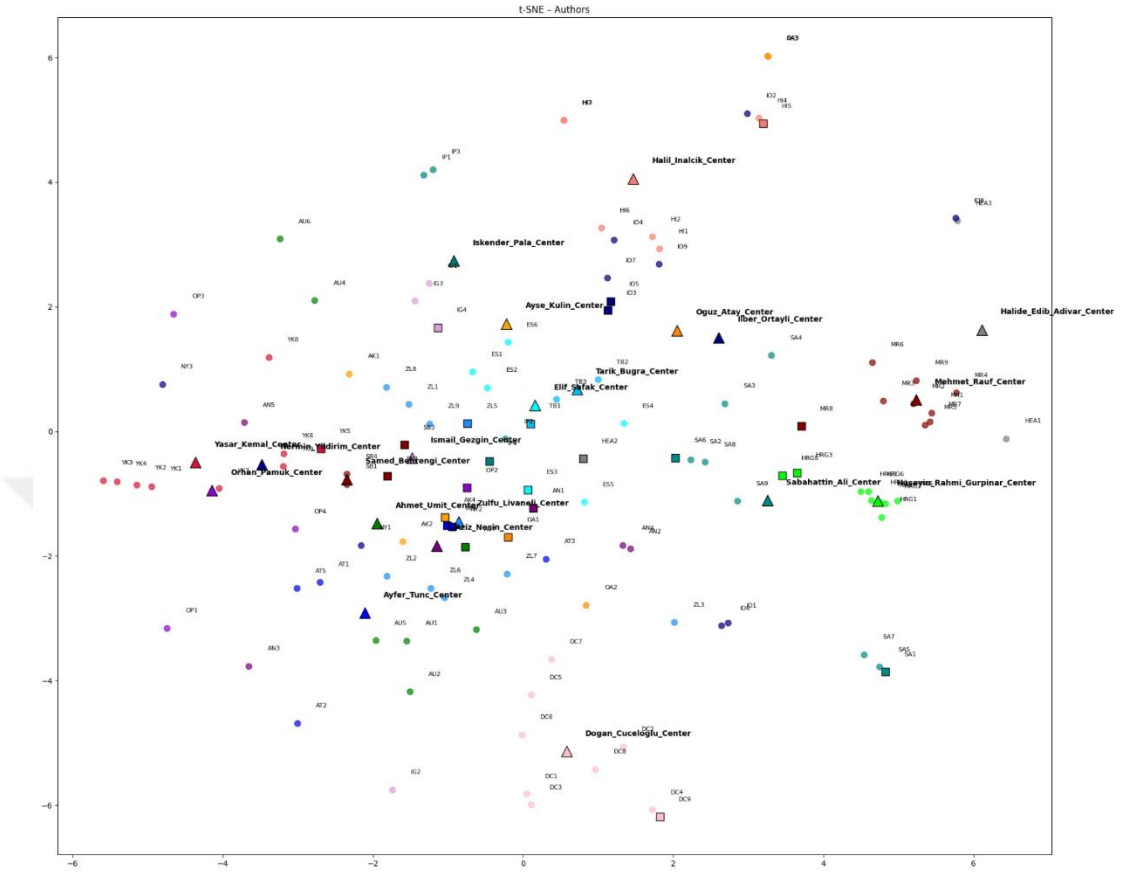

Figure 16. t-SNE visualization of Turkish book vectors grouped by author identities. Each point represents a book, and each triangle denotes the centroid of an author.

The Turkish t-SNE projection also reveals strong intra-author clustering, confirming that the model effectively learns and preserves each author's distinctive literary style and thematic focus. Notably, authors such as Hüseyin Rahmi Gürpınar, Doğan Cüceloğlu, and Sabahattin Ali form well-defined clusters, highlighting the semantic consistency within their works. Additionally, meaningful proximities are observed between authors with similar subject matter or academic backgrounds, such as Halil İnalcık and İlber Ortaylı, whose historical writings overlap in the reduced vector space. The spatial grouping of Mehmet Rauf and Hüseyin Rahmi Gürpınar further underscores the model's sensitivity to early 20th-century Ottoman-influenced prose. While these visualizations demonstrate encouraging patterns, they remain an approximation of the original high-dimensional data and should be interpreted accordingly.

5.2.5. Genre Classification Results

This subsection presents the quantitative results obtained from genre classification experiments conducted on both English and Turkish datasets. By comparing various machine learning models and the Centroid-Based Cosine Similarity approach, the analysis highlights each method's strengths and weaknesses in accurately identifying literary genres. The following parts separately discuss findings for the English and Turkish datasets, providing detailed metric-based evaluations supported by comparative tables.

5.2.5.1. With English Dataset

Table 3. English Genre Classification Results (Machine Learning and Centroid-Based Cosine Similarity)

No	Tag	Author	Book	TrueGenre	Centroid-Based Cosine Similarity	LinearSVC	LogReg	RandomForest	GaussianNB	KNN-3
1	**AC4**	**Agatha Christie**	**The Big Four**	DETE	DETE	DETE	DETE	DETE	DETE	DETE
2	**AL2**	**Andrew Lang**	**The Yellow Fairy Book**	KIDS	KIDS	KIDS	KIDS	KIDS	KIDS	KIDS
3	**AKG1**	**Anna Katharine**	**Green The Leavenworth Case**	DETE	DETE	DETE	DETE	DETE	DETE	DETE
4	**ACD1**	**Arthur Conan**	**Doyle A Study in Scarlet**	DETE	DETE	DETE	DETE	DETE	DETE	DETE
5	**CD3**	**Charles Dickens**	**A Tale of Two Cities**	CLAS	KIDS	CLAS	CLAS	CLAS	CLAS	CLAS

Table 3 (Continued). English Genre Classification Results (Machine Learning and Centroid-Based Cosine Similarity)

6	**CM2**	**Christopher Marlowe**	**Tamburlaine the Great Part1**	DRAM	DRAM	DRAM	DRAM	DRAM	DRAM	DRAM
7	**EEDS5**	**E E Doc**	**Smith Gray Lensman**	SCFI	SCFI	SCFI	SCFI	SCFI	SCFI	SCFI
8	**AP3**	**Edgar Allan**	**Poe The Murders in the Rue Morgue**	DETE	DETE	DETE	DETE	DETE	DETE	DETE
9	**ERB4**	**Edgar Rice**	**Burroughs Tarzan of the Apes**	SCFI	SCFI	SCFI	SCFI	SCFI	SCFI	SCFI
10	**EG3**	**Emile Gaboriau**	**Monsieur Lecoq**	DETE	CLAS	CLAS	CLAS	CLAS	CLAS	DETE
11	**ED2**	**Emily Dickinson**	**Poems Series 2**	POET	POET	POET	POET	POET	POET	POET
12	**FD1**	**Fyodor Dostoevsky**	**Crime and Punishment**	CLAS	CLAS	CLAS	CLAS	CLAS	CLAS	DETE
13	**GBS2**	**George Bernard**	**Shaw Arms and the Man**	DRAM	DRAM	DRAM	DRAM	DRAM	DRAM	DETE
14	**HGW2**	**H G Wells**	**The Time Machine**	SCFI	SCFI	SCFI	SCFI	DETE	DETE	DETE
15	**HI1**	**Henrik Ibsen**	**A Dolls House**	DRAM	DRAM	DRAM	DRAM	DRAM	DRAM	DRAM
16	**JA5**	**Jane Austen**	**Northanger Abbey**	CLAS	CLAS	CLAS	CLAS	CLAS	CLAS	CLAS
17	**JV3**	**Jules Verne**	**From the Earth to the Moon**	SCFI	SCFI	SCFI	SCFI	SCFI	SCFI	DETE

Table 3 (Continued). English Genre Classification Results (Machine Learning and Centroid-Based Cosine Similarity)

18	**LFB3**	**L Frank**	**Baum The Wonderful Wizard of Oz**	KIDS	KIDS	KIDS	KIDS	KIDS	KIDS	KIDS
19	**LC5**	**Lewis Carroll**	**The Hunting of the Snark**	POET	POET	POET	POET	CLAS	CLAS	POET
20	**LMA1**	**Louisa May**	**Alcott Little Women**	KIDS	KIDS	KIDS	KIDS	KIDS	KIDS	KIDS
21	**MS9**	**Mary Shelley**	**Valperga Vol3**	CLAS	CLAS	CLAS	CLAS	CLAS	CLAS	CLAS
22	**OW3**	**Oscar Wilde**	**A Woman of No Importance**	DRAM	DRAM	DRAM	DRAM	DRAM	DRAM	DRAM
23	**RAF3**	**R Austin**	**Freeman The Eye of Osiris**	DETE	DETE	DETE	DETE	DETE	DETE	DETE
24	**RF3**	**Robert Frost**	**Mountain Interval**	POET	POET	POET	POET	POET	POET	POET
25	**RLS2**	**Robert Louis**	**Stevenson Treasure Island**	CLAS	KIDS	KIDS	KIDS	CLAS	KIDS	CLAS
26	**RK4**	**Rudyard Kipling**	**Kim**	KIDS	KIDS	KIDS	KIDS	KIDS	KIDS	KIDS
27	**VH5**	**Victor Hugo**	**Toilers of the Sea**	CLAS	CLAS	CLAS	CLAS	CLAS	CLAS	DETE
28	**WC2**	**Wilkie Collins**	**The Moonstone**	CLAS	CLAS	CLAS	CLAS	CLAS	CLAS	CLAS
29	**WS6**	**William Shakespeare**	**A Midsummer Nights Dream**	DRAM	DRAM	DRAM	DRAM	DRAM	DRAM	DRAM
				ACC (%)	89.7	93.1	93.1	89.7	86.2	82.8

Table 4. Comparison of macro-averaged Accuracy, Precision, Recall, and F_1-score for English genre classification models

Model	Accuracy (%)	Precision	Recall	F_1
Centroid-Based Cosine Similarity	89.7	0.917	0.925	0.913
LinearSVC	93.1	0.943	0.948	0.943
LogReg	93.1	0.943	0.948	0.943
RandomForest	89.7	0.915	0.907	0.906
GaussianNB	86.2	0.911	0.851	0.871
KNN-3	82.8	0.924	0.836	0.849

Among the six classification approaches applied to the English dataset, LinearSVC and Logistic Regression yielded the highest overall performance across all key metrics. Both models achieved an accuracy of 93.1%, with macro precision, macro recall, and macro F_1-score all around 0.943, indicating that these methods offer robust and balanced classification capabilities for literary genre identification.

The Centroid-Based Cosine Similarity approach, while slightly behind in accuracy (89.7%), still demonstrated strong performance with a macro precision of 0.917 and a macro recall of 0.925. Given that this method is not based on traditional machine learning and has lower computational costs, its results are particularly notable and suggest its viability as a lightweight alternative in genre classification tasks.

Random Forest, although it matched the Centroid-Based Cosine Similarity method in accuracy (89.7%), showed a slightly lower macro recall (0.907 vs. 0.925). While the difference is not large, it may still indicate a minor limitation in correctly identifying certain genre types compared to similarity-based methods.

Gaussian Naive Bayes performed more modestly, with an accuracy of 86.2%. Its comparatively lower macro recall (0.851) indicates a higher likelihood of missing correct classifications, which may limit its reliability for this task.

The K-Nearest Neighbors (KNN) model, using k = 3, recorded the lowest performance among all models. With an accuracy of 82.8% and reduced scores in both macro recall

(0.836) and macro F_1-score (0.849), the model appears less capable of generalizing well in the context of genre classification. This suggests that KNN may struggle to capture the underlying semantic patterns required for effective prediction in literary texts.

In summary, LinearSVC and Logistic Regression clearly outperform the other models and emerge as the most effective methods for genre classification in this dataset. The Centroid-Based Cosine Similarity approach, despite being a non-ML baseline, offers surprisingly competitive results and demonstrates the potential of similarity-based approaches in literary analysis.

5.2.5.2. With Turkish Dataset

Table 5. Turkish Genre Classification Results (Machine Learning and Centroid-Based Cosine Similarity)

No	Tag	Author	Book	TrueGenre	Centroid-Based Cosine Similarity	LinearSVC	LogReg	RandomForest	GaussianNB	KNN-3
1	**AU7**	**Ahmet Ümit**	**Sultanı Öldürmek**	DETE	DETE	DETE	DETE	MODN	DETE	KIDS
2	**AT4**	**Ayfer Tunç**	**Suzan Defter**	MODN	MODN	MODN	MODN	MODN	MODN	KIDS
3	**AK4**	**Ayşe Kulin**	**Türkan**	BIOG	BIOG	BIOG	MODN	MODN	MODN	KIDS
4	**AN1**	**Aziz Nesin**	**Kaç Aptalız**	SOCI	SOCI	SOCI	SOCI	MODN	MODN	KIDS
5	**DC9**	**Doğan Cüceloğlu**	**Keşkesiz Bir Yaşam İçin**	PERS	PERS	PERS	PERS	PERS	PERS	PERS
6	**ES3**	**Elif Şafak**	**Med Cezir**	MODN	BIOG	MODN	MODN	MODN	MODN	KIDS

Table 5 (Continued). Turkish Genre Classification Results (Machine Learning and Centroid-Based Cosine Similarity)

7	**HEA2**	**Halide Edib Adıvar**	**Ateşten Gömlek**	CLAS	CLAS	CLAS	CLAS	MODN	CLAS	KIDS
8	**HI5**	**Halil İnalcık**	**Osmanli İmparatorluğunun Ekonomik ve Sosyal Tarihi 1**	HIST	HIST	HIST	HIST	HIST	HIST	HIST
9	**HRG3**	**Huseyin Rahmi Gürpınar**	**Mezarindan Kalkan Sehit**	SATI	SATI	SATI	SATI	SATI	CLAS	SATI
10	**HRG8**	**Huseyin Rahmi Gürpınar**	**Ölüler Yaşıyorlar mı**	SATI	SATI	SATI	SATI	SATI	CLAS	SATI
11	**IO3**	**İlber Ortaylı**	**Gazi Mustafa Kemal Ataturk**	HIST	HIST	HIST	HIST	HIST	HIST	HIST
12	**IO5**	**İlber Ortaylı**	**Osmanlida Milletler ve Diplomasi**	HIST	HIST	HIST	HIST	HIST	HIST	HIST
13	**IP4**	**İskender Pala**	**Od**	HISF	HISF	ESSA	ESSA	MODN	MODN	KIDS
14	**IG4**	**İsmail Gezgin**	**Sanatin Mitolojisi**	ESSA	ESSA	ESSA	ESSA	MODN	ESSA	KIDS
15	**MR8**	**Mehmet Rauf**	**Harabeler**	CLAS	CLAS	CLAS	CLAS	CLAS	CLAS	CLAS
16	**NY2**	**Nermin Yıldırım**	**Unutma Dersleri**	MODN	MODN	MODN	MODN	MODN	MODN	KIDS
17	**OA1**	**Oğuz Atay**	**Korkuyu Beklerken**	SHOR	MODN	MODN	MODN	MODN	MODN	KIDS
18	**OP2**	**Orhan Pamuk**	**Kara Kitap**	MODN	MODN	MODN	MODN	MODN	MODN	KIDS
19	**SA1**	**Sabahattin Ali**	**Butun Oykuleri 1**	SHOR	SHOR	SHOR	SHOR	SHOR	MODN	SHOR

Table 5 (Continued). Turkish Genre Classification Results (Machine Learning and Centroid-Based Cosine Similarity)

20	**SA6**	**Sabahattin Ali**	**İçimizdeki Seytan**	SOCI	CLAS	SOCI	SOCI	MODN	CLAS	MODN
21	**SB2**	**Samed Behrengi**	**Bir Seftali Bin Seftali**	KIDS	KIDS	KIDS	KIDS	MODN	MODN	KIDS
22	**SB3**	**Samed Behrengi**	**Kel Güvercinci**	KIDS	KIDS	KIDS	ESSA	MODN	MODN	KIDS
23	**TB1**	**Tarık Buğra**	**Firavun İmani**	CLAS	HISF	CLAS	CLAS	MODN	MODN	KIDS
24	**YK5**	**Yaşar Kemal**	**Kimsecik 3 Kanın Sesi**	MODN	KIDS	MODN	MODN	MODN	MODN	KIDS
25	**ZL5**	**Zülfü Livaneli**	**Kaplanın Sırtında**	MODN	MODN	MODN	MODN	MODN	MODN	KIDS
				ACC (%)	80.0	92.0	84.0	56.0	56.0	40.0

Table 6. Comparison of macro-averaged Accuracy, Precision, Recall, and F_1-score for Turkish genre classification models

Model	Accuracy (%)	Precision	Recall	F_1
Centroid-Based Cosine Similarity	80.0	0.844	0.861	0.822
LinearSVC	92.0	0.863	0.875	0.855
LogReg	84.0	0.757	0.750	0.724
RandomForest	56.0	0.446	0.403	0.391
GaussianNB	56.0	0.402	0.472	0.425
KNN-3	40.0	0.427	0.403	0.366

Among the six classification models tested on the Turkish dataset, LinearSVC clearly emerged as the top-performing method. With an accuracy of 92%, it achieved the highest overall success. It also outperformed the others in terms of macro precision

(0.863), macro recall (0.875), and macro F_1-score (0.855), indicating that the model was both accurate and consistent in identifying different literary genres.

Logistic Regression ranked second with an accuracy of 84%. However, its macro recall (0.750) and macro F_1-score (0.724) were noticeably lower, suggesting that the model struggled to correctly identify some genre instances. This may point to limitations in handling ambiguous or less distinctive texts.

The Centroid-Based Cosine Similarity approach, although slightly lower in accuracy (80.0%), demonstrated competitive results with macro precision (0.844) and macro recall (0.861). Given its unsupervised nature and minimal computational requirements, it proves to be a strong alternative for genre classification, especially in scenarios with limited resources.

In contrast, both the Random Forest and Gaussian Naive Bayes models performed poorly, each with an accuracy of 56.0%. Their macro F_1-scores (0.391 and 0.425 respectively) were significantly lower than the leading methods. These results suggest that such models may struggle to capture the nuanced linguistic and structural features that distinguish literary genres.

The K-Nearest Neighbors (KNN) model showed the weakest performance overall, with an accuracy of only 40.0% and a macro F_1-score of 0.366. These findings indicate that KNN may not be suitable for complex text classification tasks that require sensitivity to deeper contextual cues.

In conclusion, LinearSVC demonstrated the most effective and consistent results on the Turkish dataset. Interestingly, the Centroid-Based Cosine Similarity model, despite its simplicity, delivered surprisingly competitive performance. The remaining models, especially those with less robust learning mechanisms, struggled to correctly classify genres, particularly when the structural boundaries between genres were less pronounced.

5.2.6. Author Classification Results

This subsection discusses the quantitative results obtained from author classification experiments on both English and Turkish datasets. The findings illustrate how different classification models perform in capturing authorial style and semantic patterns, offering a comparative perspective on their strengths and weaknesses.

5.2.6.1. With English Dataset

Table 7. English Author Classification Results (Machine Learning and Centroid-Based Cosine Similarity)

No	Tag	Author	Book	Centroid-Based Cosine Similarity	LinearSVC	LogReg	RandomForest	GaussianNB	KNN-3
1	**AC4**	**Agatha Christie**	**The Big Four**	AC	AC	AC	AC	EAP	AC
2	**AL2**	**Andrew Lang**	**The Yellow Fairy Book**	AL	AL	AL	AL	OW	AL
3	**AKG1**	**Anna Katharine Green**	**The Leavenworth Case**	AKG	AKG	AKG	AKG	AKG	AKG
4	**ACD1**	**Arthur Conan Doyle**	**A Study in Scarlet**	ACD	ACD	ACD	ACD	ACD	ACD
5	**CD3**	**Charles Dickens**	**A Tale of Two Cities**	CD	CD	CD	CD	CD	CD
6	**CM2**	**Christopher Marlowe**	**Tamburlaine I**	CM	CM	CM	CM	WS	CM
7	**EEDS5**	**E E Doc Smith**	**Gray Lensman**	EEDS	EEDS	EEDS	EEDS	EEDS	EEDS
8	**AP3**	**Edgar Allan Poe**	**Rue Morgue**	EAP	EAP	EAP	EAP	EAP	AKG

Table 7 (Continued). English Author Classification Results (Machine Learning and Centroid-Based Cosine Similarity)

9	**ERB4**	**Edgar Rice Burroughs**	**Tarzan of the Apes**	ERB	ERB	ERB	ERB	RK	ERB
10	**EG3**	**Emile Gaboriau**	**Monsieur Lecoq**	EG	EG	EG	FD	EAP	ACD
11	**ED2**	**Emily Dickinson**	**Poems (Series 2)**	ED	ED	ED	ED	EAP	ED
12	**FD1**	**Fyodor Dostoevsky**	**Crime and Punishment**	FD	FD	FD	FD	FD	ACD
13	**GBS2**	**George Bernard Shaw**	**Arms and the Man**	GBS	GBS	GBS	HI1	OW	ACD
14	**HGW2**	**H G Wells**	**The Time Machine**	HGW	HGW	HGW	HGW	HGW	ACD
15	**HI1**	**Henrik Ibsen**	**A Doll's House**	HI1	HI1	HI1	HI1	OW	HI1
16	**JA5**	**Jane Austen**	**Northanger Abbey**	JA	JA	JA	JA	CD	JA
17	**JV3**	**Jules Verne**	**From the Earth to the Moon**	JV	VH	VH	VH	VH	ACD
18	**LFB3**	**L Frank Baum**	**Wizard of Oz**	LFB	LFB	LFB	LFB	RK	LFB
19	**LC5**	**Lewis Carroll**	**The Hunting of the Snark**	ED	LC	LC	LC	EAP	ED
20	**LMA1**	**Louisa May Alcott**	**Little Women**	LMA	LMA	LMA	LMA	CD	LMA
21	**MS9**	**Mary Shelley**	**Valperga (V3)**	MS	MS	MS	MS	MS	MS
22	**OW3**	**Oscar Wilde**	**A Woman of No Importance**	OW	OW	OW	OW	OW	OW
23	**RAF3**	**R Austin Freeman**	**The Eye of Osiris**	RAF	RAF	RAF	ACD	EAP	AC

Table 7 (Continued). English Author Classification Results (Machine Learning and Centroid-Based Cosine Similarity)

24	**RF3**	**Robert Frost**	**Mountain Interval**	RF	RF	RF	RF	RK	RF
25	**RLS2**	**Robert Louis Stevenson**	**Treasure Island**	RLS	RLS	RLS	RK	CD	ACD
26	**RK4**	**Rudyard Kipling**	**Kim**	RK	RK	RK	RK	RK	RK
27	**VH5**	**Victor Hugo**	**Toilers of the Sea**	VH	VH	VH	VH	VH	ACD
28	**WC2**	**Wilkie Collins**	**The Moonstone**	CD	WC	CD	CD	CD	ACD
29	**WS6**	**William Shakespeare**	**Midsummer Night's Dream**	WS	WS	WS	WS	WS	WS
	ACC (%)			93.1	96.6	93.1	79.3	41.4	62.1

Table 8. Comparison of macro-averaged Accuracy, Precision, Recall, and F_1-score for English author classification models

Model	**Accuracy (%)**	**Precision**	**Recall**	F_1
Centroid-Based Cosine Similarity	93.1	0.897	0.931	0.908
LinearSVC	96.6	0.948	0.966	0.954
LogReg	93.1	0.897	0.931	0.908
RandomForest	79.3	0.690	0.793	0.724
GaussianNB	41.4	0.271	0.414	0.302
KNN-3	62.1	0.538	0.621	0.559

In the author classification task conducted on the English book dataset, LinearSVC clearly emerged as the top-performing model. It achieved the highest accuracy of 96.6%, with strong results in macro precision (0.948), macro recall (0.966), and macro F_1-score (0.954). These results indicate that the model is highly effective at identifying the stylistic patterns and linguistic features specific to each author.

Logistic Regression and the Centroid-Based Cosine Similarity approach followed closely behind, both achieving an accuracy of 93.1%. Logistic Regression matched the centroid-based method in macro precision (0.897), macro recall (0.931), and macro F_1-score (0.908), demonstrating reliable performance across all metrics. The Centroid-Based Cosine Similarity approach, despite not involving any training phase, proves to be a strong alternative due to its simplicity and computational efficiency, making it particularly attractive in resource-constrained environments.

Random Forest delivered a significantly lower accuracy of 79.3%, with a macro F_1-score of 0.724 and macro recall of 0.793. These results suggest that it may struggle to generalize across diverse writing styles, leading to inconsistent author predictions.

Gaussian Naive Bayes performed poorly, achieving only 41.4% accuracy. Its macro F_1-score (0.302) and macro recall (0.414) reflect a high rate of misclassification, likely due to its overly simplistic assumptions about feature independence.

The weakest performance was recorded by the K-Nearest Neighbors (KNN-3) model, with an accuracy of 62.1% and a macro F_1-score of 0.559. These results suggest that KNN is ill-suited for capturing the fine-grained distinctions in authorial style required for this task.

In conclusion, LinearSVC stands out as the most reliable method for authorship classification in this dataset. Meanwhile, the Centroid-Based Cosine Similarity approach remains a lightweight and interpretable alternative that still delivers respectable performance.

5.2.6.2. With Turkish Dataset

Table 9. Turkish Author Classification Results (Machine Learning and Centroid-Based Cosine Similarity)

No	Tag	Author	Book	Centroid-Based Cosine Similarity	LinearSVC	LogReg	RandomForest	GaussianNB	KNN-3
1	**AU7**	**Ahmet Ümit**	**Sultani Oldurmek**	AU	AU	AU	ZL	AU	SB
2	**AT4**	**Ayfer Tunç**	**Suzan Defter**	AT	AT	AT	AT	ZL	SB
3	**AK4**	**Ayşe Kulin**	**Turkan**	AK	AK	AK	ZL	ZL	SB
4	**AN1**	**Aziz Nesin**	**Kac Aptaliz**	AN	AN	AN	SA	AN	SB
5	**DC9**	**Doğan Cüceloğlu**	**Keskesiz Bir Yasam Icin**	DC	DC	DC	DC	DC	DC
6	**ES3**	**Elif Şafak**	**Med Cezir**	ES	ES	ES	ZL	ZL	SB
7	**HEA2**	**Halide Edib Adıvar**	**Adivar Atesten Gomlek**	MR	SA	MR	SA	ZL	SB
8	**HI5**	**Halil İnalcık**	**Osmanli Imparatorlugunun Ekonomik ve Sosyal Tarihi 1**	HI	HI	HI	HI	HI	AK
9	**HRG3**	**Hüseyin Rahmi Gürpınar**	**Gurpinar Mezarindan Kalkan Sehit**	HRG	HRG	HRG	HRG	SA	HRG

Table 9 (Continued). Turkish Author Classification Results (Machine Learning and Centroid-Based Cosine Similarity)

10	**HRG8**	**Hüseyin Rahmi Gürpınar**	**Gurpinar Oluler Yasiyorlar mi**	HRG	HRG	HRG	SA	SA	HRG
11	**IO3**	**İlber Ortaylı**	**Gazi Mustafa Kemal Ataturk**	IO	IO	IO	IO	IO	IO
12	**IO5**	**İlber Ortaylı**	**Osmanlida Milletler ve Diplomasi**	IO	IO	IO	IO	IO	IO
13	**IP4**	**İskender Pala**	**Od**	IP	IP	IP	ZL	ZL	SB
14	**IG4**	**İsmail Gezgin**	**Sanatin Mitolojisi**	IG	IG	IG	IG	ZL	SB
15	**MR8**	**Mehmet Rauf**	**Harabeler**	MR	MR	MR	SA	MR	MR
16	**NY2**	**Nermin Yıldırım**	**Unutma Dersleri**	NY	NY	NY	ZL	ZL	SB
17	**OA1**	**Oğuz Atay**	**Korkuyu Beklerken**	OA	OA	OA	ZL	AU	SB
18	**OP2**	**Orhan Pamuk**	**Kara Kitap**	OP	OP	OP	ZL	ZL	SB
19	**SA1**	**Sabahattin Ali**	**Butun Oykuleri1**	SA	SA	SA	SA	SA	SA
20	**SA6**	**Sabahattin Ali**	**Icimizdeki Seytan**	SA	SA	SA	SA	SA	SA
21	**SB2**	**Samed Behrengi**	**Bir Seftali Bin Seftali**	SB	SB	SB	SB	YK	SB
22	**SB3**	**Samed Behrengi**	**Kel Guvercinci**	SB	SB	SB	SB	YK	SB

Table 9 (Continued). Turkish Author Classification Results (Machine Learning and Centroid-Based Cosine Similarity)

23	**TB1**	**Tarık Buğra**	**Firavun Imani**	TB	TB	TB	ZL	ZL	SB
24	**YK5**	**Yaşar Kemal**	**Kimsecik3 Kanin Sesi**	YK	YK	YK	YK	YK	SB
25	**ZL5**	**Zülfü Livaneli**	**Kaplanin Sirtinda**	ZL	ZL	ZL	ZL	ZL	SB
		ACC (%)		96.0	96.0	96.0	52.0	44.0	40.0

Table 10. Comparison of macro-averaged Accuracy, Precision, Recall, and F_1-score for Turkish author classification models

Model	**Accuracy (%)**	**Precision**	**Recall**	F_1
Centroid-Based Cosine Similarity	96.0	0.929	0.952	0.937
LinearSVC	96.0	0.937	0.952	0.943
LogReg	96.0	0.929	0.952	0.937
RandomForest	52.0	0.402	0.452	0.398
GaussianNB	44.0	0.306	0.429	0.334
KNN-3	40.0	0.244	0.286	0.249

In the author classification experiments conducted on the Turkish book dataset, three models (Centroid-Based Cosine Similarity approach, LinearSVC, and Logistic Regression) achieved equally high performance, each reaching an accuracy of 96.0%. However, a closer inspection of the evaluation metrics reveals that LinearSVC offered the most consistent and balanced performance, with a macro precision of 0.937 and a macro F_1-score of 0.943.

Both the Centroid-Based Cosine Similarity approach and Logistic Regression delivered impressive results with the same accuracy and identical macro F_1-score values (0.937). Particularly notable is the performance of the cosine-based method, which, despite not requiring any training, was able to capture authorial similarities effectively based solely on vector proximity to the class centroids. This demonstrates its potential as a lightweight yet powerful baseline method for authorship attribution.

While these three models achieved identical accuracy scores, Random Forest performed considerably worse, with an accuracy of 52.0% and a macro F_1-score of 0.398. Its randomized decision-making process may also lead to reduced interpretability and less consistency across different runs.

On the other hand, both Gaussian Naive Bayes and K-Nearest Neighbors (KNN-3) performed poorly. GaussianNB reached only 44.0% accuracy, while KNN dropped further to 40.0%, with both models showing significantly lower macro precision and macro F_1-scores. The inability of KNN, in particular, to capture the stylistic and semantic nuances of authorial language appears to have hindered its effectiveness in this task.

These results suggest that for author classification based on long-form textual data, models that rely on vector space similarity (such as the Centroid-Based Cosine Similarity approach) or linear decision boundaries (such as SVC and Logistic Regression) are far more effective than simpler probabilistic or instance-based approaches.

5.2.7. General Observations and Insights

The comparative analysis of classification models across both English and Turkish datasets reveals several consistent trends regarding the effectiveness of different algorithmic approaches in literary text classification tasks. Linear Support Vector Classification (LinearSVC) and Logistic Regression emerged as the most robust and reliable models across both genre and author identification tasks, demonstrating consistently high scores in accuracy, precision, recall, and F_1. These models exhibited strong generalization capabilities in distinguishing stylistic and thematic patterns,

regardless of the language or text complexity. Notably, the Centroid-Based Cosine Similarity approach, although not based on supervised learning, yielded competitive results, especially in the Turkish dataset, where it occasionally matched or even outperformed some machine learning models. This highlights the potential of unsupervised, vector-based approaches in tasks where computational efficiency and simplicity are prioritized. In contrast, models such as K-Nearest Neighbors (KNN) and Gaussian Naive Bayes (GNB) generally underperformed, particularly in tasks requiring the discrimination of subtle linguistic nuances across authors or genres. Overall, the findings underscore the importance of selecting models that balance accuracy with interpretability and computational cost, with LinearSVC, Logistic Regression, and the Centroid-Based Cosine Similarity approach offering the most promising trade-offs for practical NLP applications in literary analysis.

5.2.8. Out-of-Sample Evaluation

Out-of-sample evaluation is a critical method used to assess the generalization ability of a machine learning model. In this approach, the model is tested on authors and books it has never encountered during training. This allows for a more accurate analysis of the model's ability to make correct predictions on entirely new data. Out-of-sample evaluation simulates real-world scenarios where the model must classify texts from unknown sources. In this thesis, both the English and Turkish datasets were evaluated separately under this framework for the genre classification task.

Table 11. English Genre Classification Results - Out-of-Sample Evaluation (Machine Learning and Centroid-Based Cosine Similarity)

No	Author Book Count	Author	Book	Tag	Split	Genre
1	1	Anna Sewell	Black Beauty	AS_T	Test	Children and Young Adult
2	1	Brothers Grimm	Grimms Fairy Tales	BG_T	Test	Children and Young Adult
3	1	Daniel Defoe	Robinson Crusoe	DD_T	Test	Classic Novels
4	1	Dorothy L. Sayers	Whose Body	DLS_T	Test	Detective Novels
5	1	Hugh Lofting	The Story of Doctor Dolittle	HL_T	Test	Children and Young Adult
6	1	J M	Barrie Peter Pan	JB_T	Test	Children and Young Adult
7	1	Jack London	The Scarlet Plague	JL_T	Test	Science Fiction
8	1	Jonathan Swift	Gullivers Travels	JS_T	Test	Classic Novels
9	1	Kenneth Grahame	The Wind in the Willows	KG_T	Test	Children and Young Adult
10	1	Mark Twain	Tom Sawyer Abroad	MT_T	Test	Children and Young Adult
11	1	Mary Elizabeth Braddon	Lady Audley's Secret	MEB_T	Test	Detective Novels
12	1	Moliere Tartuffe	Moliere Tartuffe	MOL_T	Test	Drama Tiyatro
13	1	Samuel Butler	Erewhon	SBT_T	Test	Science Fiction

Table 11 (Continued). English Genre Classification Results - Out-of-Sample Evaluation (Machine Learning and Centroid-Based Cosine Similarity)

14	1	Samuel Taylor	Coleridge The Rime of the Ancient Mariner	STC_T	Test	Poetry
15	2	Walt Whitman	Leaves of Grass	WW1_T	Test	Poetry
16		Walt Whitman	Drum Taps	WW2_T	Test	Poetry
17	2	William Blake	Songs of Innocence and Experience	WB1_T	Test	Poetry
18		William Blake	The Marriage of Heaven and Hell	WB2_T	Test	Poetry

The out-of-sample evaluation subset of the English dataset consists of 18 books written by 16 different authors. These books are distributed across six literary genres: 6 books belong to the "Children and Young Adult" genre, 2 to "Classic Novels," 2 to "Detective Novels," 1 to "Drama," 5 to "Poetry," and 2 to "Science Fiction." This set includes works by well-known authors such as Mary Elizabeth Braddon, William Blake, Jack London, and Jonathan Swift. With its wide range of styles and historical periods, this subset provides a robust ground for evaluating how well the model can generalize across different genres and authorial voices.

Table 12. Turkish Genre Classification Results - Out-of-Sample Evaluation (Machine Learning and Centroid-Based Cosine Similarity)

No	Author Book Count	Author	Book	Tag	Split	Genre
1	1	Ahmet Şerif İzgören	Avucunuzdaki Kelebek	ASI_T	Test	Personal Development
2	1	Aydın Çetiner	21. Yüzyılda Savaş Stratejileri	AC_T	Test	History
3	1	Büşra Sanay	Yaşadım Demek İçin Ne Yapmalı	BS_T	Test	Personal Development
4	1	Canan Tan	Sızı	CT_T	Test	Modern Novel
5	1	Cansu Canan Özgen	Türklerin Serüveni	CCO_T	Test	History
6	1	Erhan Afyoncu	Osmanlı'da Askeri İsyanlar ve Darbeler	EA_T	Test	History
7	1	Ipek Ongun	Mektup Arkadaslari	ION_T	Test	Children Youth
8	1	Merve Özcan	Gözlerini Haramdan Sakın	MO_T	Test	Modern Novel
9	1	Mümin Sekman	Her Şey Seninle Başlar	MSK_T	Test	Personal Development
10	2	Nurdan Gürbilek	Kötü Çocuk Türk	NGKC_T	Test	Essay Culture
11		Nurdan Gürbilek	Kör Ayna Kayıp Şark	NGK_T	Test	Essay Culture
12	1	Peyami Safa	Yalnızız	PS_T	Test	Social Critique
13	1	Resat Nuri	Guntekin Calikusu	RNGC_T	Test	Classic Novel

Table 12 (Continued). Turkish Genre Classification Results - Out-of-Sample Evaluation (Machine Learning and Centroid-Based Cosine Similarity)

14	1	Râşid Erer	Türklere Karşı Haçlı Seferleri	RE_T	Test	History
15	1	Rıfat Ilgaz	Sosyal Kadınlar Partisi	RI3_T	Test	Satire Humour
16	1	Sait Faik Abasıyanık	Havuz Bası, Son Kuşlar	SFAH_T	Test	Short Story
17	1	Samipasazade Sezai	Serguzest	SS_T	Test	Classic Novel
18	1	Sevinç Yavuz	Türk Seri Katiller	SY_T	Test	Detective Novel
19	1	Tevfik Guran	19 Yuzyilda Osmanli Ekonomisi Uzerine Arastirmalar	TG_T	Test	History
20	1	Yakup Kadri	Karaosmanoglu Kiralik Konak	YKK_T	Test	Classic Novel
21	1	İhsan Oktay Anar	Puslu Kıtalar Atlası	IOA_T	Test	Historical Fiction
22	1	Şevket Süreyya Aydemir	Suyu Arayan Adam	SSA_T	Test	Biography Memoir

The Turkish out-of-sample dataset comprises 22 books by 21 unique authors, each contributing only one work to the evaluation set. These books span 12 distinct genres: 3 books fall under “Personal Development,” 5 under “History,” 2 under “Modern Novel,” 1 under “Children and Youth,” 2 under “Essay and Culture,” 1 under “Social Critique,” 3 under “Classic Novel,” 1 under “Satire and Humour,” 1 under “Short Story,” 1 under “Detective Novel,” 1 under “Historical Fiction,” and 1 under “Biography and Memoir.” The diversity of this evaluation set, encompassing both fiction and non-fiction texts, makes it a valuable benchmark for testing the model’s multi-genre classification capabilities.

5.2.8.1. Genre Prediction - English (Unseen Authors)

Table 13. English Genre Classification Results - Out-of-Sample Evaluation (Machine Learning and Centroid-Based Cosine Similarity)

No	Tag	Author	Book	TrueGenre	Centroid-Based Cosine Similarity	LinearSVC	LogReg	RandomForest	GaussianNB	KNN-3
1	**DD_T**	**Daniel Defoe**	**Robinson Crusoe**	CLAS	CLAS	SCFI	CLAS	CLAS	CLAS	CLAS
2	**JS_T**	**Jonathan Swift**	**Gulliver's Travels**	CLAS	CLAS	CLAS	CLAS	CLAS	CLAS	DETE
3	**DLS_T**	**Dorothy L. Sayers**	**Whose Body**	DETE	DETE	DETE	DETE	DETE	DETE	DETE
4	**MEB_T**	**Mary Elizabeth Braddon**	**Lady Audley's Secret**	DETE	CLAS	CLAS	CLAS	CLAS	CLAS	CLAS
5	**AS_T**	**Anna Sewell**	**Black Beauty**	KIDS	KIDS	KIDS	KIDS	KIDS	KIDS	KIDS
6	**BG_T**	**Brothers Grimm**	**Grimm's Fairy Tales**	KIDS	KIDS	KIDS	KIDS	KIDS	KIDS	KIDS
7	**HL_T**	**Hugh Lofting**	**The Story of Doctor Dolittle**	KIDS	KIDS	KIDS	KIDS	KIDS	KIDS	KIDS
8	**JB_T**	**J. M. Barrie**	**Peter Pan**	KIDS	KIDS	KIDS	KIDS	CLAS	KIDS	CLAS
9	**KG_T**	**Kenneth Grahame**	**The Wind in the Willows**	KIDS	KIDS	KIDS	KIDS	KIDS	KIDS	KIDS

Table 13 (Continued). English Genre Classification Results - Out-of-Sample Evaluation (Machine Learning and Centroid-Based Cosine Similarity)

10	**MT_T**	**Mark Twain**	**Tom Sawyer Abroad**	KIDS	KIDS	KIDS	KIDS	KIDS	KIDS	POET
11	**MOL_T**	**Molière**	**Tartuffe**	DRAM	DRAM	DRAM	DRAM	DRAM	DRAM	DETE
12	**WW1_T**	**Walt Whitman**	**Leaves of Grass**	POET	POET	POET	POET	DRAM	DRAM	POET
13	**WW2_T**	**Walt Whitman**	**Drum Taps**	POET	POET	POET	POET	CLAS	DRAM	POET
14	**WB1_T**	**William Blake**	**Songs of Innocence and Experience**	POET	POET	POET	POET	POET	POET	POET
15	**WB2_T**	**William Blake**	**The Marriage of Heaven and Hell**	POET	POET	POET	POET	CLAS	POET	POET
16	**STC_T**	**Samuel Taylor Coleridge**	**The Rime of the Ancient Mariner**	POET	POET	POET	POET	POET	POET	POET
17	**JL_T**	**Jack London**	**The Scarlet Plague**	SCFI	SCFI	SCFI	SCFI	KIDS	KIDS	POET
18	**SBT_T**	**Samuel Butler**	**Erewhon**	SCFI	SCFI	SCFI	SCFI	CLAS	SCFI	CLAS
ACC (%)					94.4	88.9	94.4	61.1	77.8	61.1

Table 14. Comparison of macro-averaged Accuracy, Precision, Recall, and F_1-score for English genre classification models – Out-of-Sample Evaluation

Model	Accuracy (%)	Precision	Recall	F_1
Centroid-Based Cosine Similarity	94.4	0.944	0.917	0.911
LinearSVC	88.9	0.861	0.833	0.828
LogReg	94.4	0.944	0.917	0.911
RandomForest	61.1	0.603	0.622	0.530
GaussianNB	77.8	0.810	0.767	0.718
KNN-3	61.1	0.383	0.444	0.394

This table presents the genre classification results for the English dataset, specifically focusing on books written by unseen authors, that is, authors whose works were not included in the training phase. This type of out-of-sample evaluation is crucial for measuring a model's generalization capability, as it simulates real-world scenarios where the model must classify newly encountered texts.

Logistic Regression and the Centroid-Based Cosine Similarity approach emerged as the top performers, both achieving an accuracy of 94.4%, with macro precision and macro recall values of 0.944 and 0.917, respectively. Their macro F_1-score of 0.911 further confirms their robustness. Particularly noteworthy is the success of the Centroid-Based Cosine Similarity method, which, despite not requiring any training, managed to rival a supervised model like Logistic Regression. This reinforces its value as a simple yet effective baseline for genre classification.

LinearSVC also demonstrated strong performance with an accuracy of 88.9% and a macro F_1-score of 0.828. Although slightly behind the top two models, LinearSVC still maintained a solid balance between macro precision (0.861) and macro recall (0.833), showcasing its ability to generalize effectively to unfamiliar texts. As a supervised model, its structured decision boundaries appear to transfer reasonably well to new data, making it a reliable and interpretable choice in practical applications.

Gaussian Naive Bayes, with an accuracy of 77.8%, yielded moderate results. Although its macro precision (0.810) and macro recall (0.767) are fairly balanced, the lower macro F_1-score (0.718) suggests some inconsistency, likely due to its simplistic probabilistic assumptions.

In contrast, Random Forest and K-Nearest Neighbors (KNN-3) underperformed. Both recorded an accuracy of only 61.1%, with KNN displaying particularly poor macro precision (0.383) and macro F_1-score (0.394), indicating frequent misclassifications. These models struggled significantly with capturing the semantic and stylistic features needed to classify books from previously unseen authors.

In summary, Logistic Regression, the Centroid-Based Cosine Similarity approach, and LinearSVC stand out as the most effective methods for genre classification when dealing with unseen authors. Their strong generalization capabilities highlight the potential of both supervised and unsupervised approaches in NLP-driven literary analysis.

5.2.8.2. Genre Prediction - Turkish (Unseen Authors)

Table 15. Turkish Genre Prediction - Unseen Authors (Machine Learning and Centroid-Based Cosine Similarity)

No	Tag	Author	Book	TrueGenre	Centroid-Based Cosine Similarity	LinearSVC	LogReg	RandomForest	GaussianNB	KNN-3
1	**SS_T**	**Samipaşazade Sezai**	**Sergüzeşt**	CLAS	CLAS	CLAS	CLAS	MODN	CLAS	KIDS
2	**YKK_T**	**Yakup Kadri Karaosmanoğlu**	**Kiralık Konak**	CLAS	CLAS	CLAS	CLAS	MODN	CLAS	SATI
3	**RNGC_T**	**Reşat Nuri Güntekin**	**Çalıkuşu**	CLAS	CLAS	CLAS	CLAS	MODN	CLAS	KIDS
4	**CT_T**	**Canan Tan**	**Sızı**	MODN	MODN	MODN	MODN	MODN	MODN	KIDS

Table 15 (Continued). Turkish Genre Prediction - Unseen Authors (Machine Learning and Centroid-Based Cosine Similarity)

5	**MO_T**	**Merve Özcan**	**Gözlerini Haramdan Sakın**	MODN	DETE	MODN	MODN	MODN	MODN	KIDS
6	**SY_T**	**Sevinç Yavuz**	**Türk Seri Katiller**	DETE	DETE	DETE	DETE	MODN	MODN	KIDS
7	**SFAH_T**	**Sait Faik Abasıyanık**	**Havuz Başı, Son Kuşlar**	SHOR	HIST	HIST	HIST	MODN	MODN	KIDS
8	**TG_T**	**Tevfik Güran**	**19. Yüzyılda Osmanlı Ekonomisi Üzerine Araştırmalar**	HIST	HIST	HIST	HIST	MODN	HIST	KIDS
9	**AC_T**	**Aydın Çetiner**	**21. Yüzyılda Savaş Stratejileri**	HIST	HIST	HIST	HIST	MODN	HIST	KIDS
10	**CCO_T**	**Cansu Canan Özgen**	**Türklerin Serüveni**	HIST	HIST	HIST	HIST	MODN	HIST	KIDS
11	**EA_T**	**Erhan Afyoncu**	**Osmanlı'da Askerî İsyanlar ve Darbeler**	HIST	HIST	HIST	HIST	MODN	HIST	KIDS
12	**RE_T**	**Râşid Erer**	**Türklere Karşı Haçlı Seferleri**	HIST	HIST	HIST	HIST	MODN	HIST	KIDS
13	**ASI_T**	**Ahmet Şerif**	**İzgören Avucunuzdaki Kelebek**	PERS	PERS	PERS	PERS	MODN	PERS	KIDS
14	**BS_T**	**Büşra Sanay**	**Yaşadım Demek İçin Ne Yapmalı**	PERS	PERS	PERS	PERS	MODN	PERS	KIDS
15	**MSK_T**	**Mümin Sekman**	**Her Şey Seninle Başlar**	PERS	PERS	PERS	PERS	MODN	PERS	KIDS
16	**ION_T**	**İpek Ongun**	**Mektup Arkadaşları**	KIDS	KIDS	KIDS	KIDS	MODN	MODN	KIDS

Table 15 (Continued). Turkish Genre Prediction - Unseen Authors (Machine Learning and Centroid-Based Cosine Similarity)

17	**SSA_T**	**Şevket Süreyya**	**Aydemir Suyu Arayan Adam**	BIOG	CLAS	HIST	CLAS	MODN	MODN	KIDS
18	**IOA_T**	**İhsan Oktay**	**Anar Puslu Kıtalar Atlası**	HISF	MODN	MODN	MODN	MODN	MODN	KIDS
19	**NGK_T**	**Nurdan Gürbilek**	**Kör Ayna Kayıp Şark**	ESSA	ESSA	ESSA	ESSA	MODN	MODN	SATI
20	**NGKC_T**	**Nurdan Gürbilek**	**Kötü Çocuk Türk**	ESSA	ESSA	ESSA	ESSA	MODN	MODN	KIDS
21	**PS_T**	**Peyami Safa**	**Yalnızız**	SOCI	SATI	CLAS	CLAS	MODN	MODN	SATI
22	**RI3_T**	**Rıfat Ilgaz**	**Sosyal Kadınlar Partisi**	SATI	HIST	HIST	HIST	MODN	MODN	KIDS
ACC (%)					72.7	77.3	77.3	9.1	59.1	4.5

Table 16. Comparison of macro-averaged Accuracy, Precision, Recall, and F_1-score for Turkish genre classification models – Unseen Authors

Model	**Accuracy (%)**	**Precision**	**Recall**	***F*1**
Centroid-Based Cosine Similarity	72.7	0.455	0.542	0.488
LinearSVC	77.3	0.503	0.583	0.536
LogReg	77.3	0.498	0.583	0.532
RandomForest	9.1	0.008	0.083	0.014
GaussianNB	59.1	0.265	0.333	0.276
KNN-3	4.5	0.004	0.083	0.008

This table presents the genre prediction results for previously unseen Turkish authors, serving as an out-of-sample evaluation to test how well the models generalize to completely new writers. Such evaluations are critical for assessing the robustness of classification models when exposed to literary works outside the training distribution.

In this experiment, LinearSVC and Logistic Regression were the top-performing models, both achieving an accuracy of 77.3%. While this score is somewhat lower than those observed in previous experiments on familiar authors (which typically ranged between 85% and 95%), it still represents a reasonable level of performance in the context of open-set genre classification. The macro F_1-scores of 0.536 and 0.532, respectively, indicate that these models retained the ability to generalize to new authors with acceptable reliability. Notably, LinearSVC once again demonstrated its strength in handling stylistic and thematic patterns across unfamiliar content.

The Centroid-Based Cosine Similarity approach, despite not requiring any training, yielded a competitive accuracy of 72.7% and a macro F_1-score of 0.488. This result supports its viability as a lightweight, efficient alternative, especially in cases where computational simplicity and fast inference are priorities. It managed to capture meaningful genre-related patterns based solely on vector space similarity.

Gaussian Naive Bayes performed moderately with 59.1% accuracy and a macro F_1-score of 0.276, suggesting its limited capability in distinguishing complex genre features in texts from unseen authors. Its probabilistic assumptions may not be adequate for capturing the subtle variations in linguistic or thematic style.

By contrast, Random Forest and K-Nearest Neighbors (KNN-3) performed very poorly. Random Forest scored only 9.1% accuracy, and KNN-3 just 4.5%, with macro F_1-scores of 0.014 and 0.008 respectively, indicating that these models largely failed to generalize to new authors. This poor performance suggests that both models are heavily dependent on the training data distribution and lack the flexibility needed to adapt to unfamiliar patterns in new texts.

It is also worth noting that certain genres, such as history and personal development, were predicted more consistently, whereas genres like satire, humour, social criticism, and biography appeared more challenging when applied to previously unseen authors. This discrepancy may indicate that exposure to a wider and more diverse training

dataset could enhance the model's ability to capture the nuances of such genres more effectively.

In summary, LinearSVC and Logistic Regression once again stood out as the most reliable classifiers, maintaining stable performance even under challenging out-of-sample conditions. Meanwhile, the Centroid-Based Cosine Similarity approach proved to be an effective non-ML benchmark. Simpler models like Random Forest and KNN, which rely on memorization or local patterns, were unable to cope with the diversity and complexity introduced by unseen authors in the Turkish literary domain.

5.2.8.3. Author Prediction - English (Unseen Authors)

Table 17. English Author Prediction - Unseen Authors (Machine Learning and Centroid-Based Cosine Similarity)

No	Tag	Author	Book	Centroid-Based Cosine Similarity	LinearSVC	LogReg	RandomForest	GaussianNB	KNN-3
1	**DD_T**	**Daniel Defoe**	**Robinson Crusoe**	JV	JV	JV	CD	RK	CD
2	**JS_T**	**Jonathan Swift**	**Gulliver's Travels**	CM	LFB	VH	MS	MS	ACD
3	**DLS_T**	**Dorothy L. Sayers**	**Whose Body**	AC	AC	AC	AC	OW	ACD
4	**MEB_T**	**Mary Elizabeth Braddon**	**Lady Audley's Secret**	CD	CD	CD	CD	OW	ACD
5	**AS_T**	**Anna Sewell**	**Black Beauty**	RK	RK	RK	CD	RK	LMA
6	**BG_T**	**Brothers Grimm**	**Grimm's Fairy Tales**	AL	AL	AL	AL	RK	AL
7	**HL_T**	**Hugh Lofting**	**The Story of Doctor Dolittle**	RK	RK	RK	RK	RK	LC

Table 17 (Continued). English Author Prediction - Unseen Authors (Machine Learning and Centroid-Based Cosine Similarity)

8	**JB_T**	**J. M. Barrie**	**Peter Pan**	CD	CD	CD	CD	OW	CD
9	**KG_T**	**Kenneth Grahame**	**The Wind in the Willows**	RK	RK	RK	RK	RK	ACD
10	**MT_T**	**Mark Twain**	**Tom Sawyer Abroad**	RK	RK	RK	RK	RK	RF
11	**MOL_T**	**Molière**	**Tartuffe**	WS	WS	WS	OW	OW	ACD
12	**WW1_T**	**Walt Whitman**	**Leaves of Grass**	CM	CM	CM	RF	RK	RF
13	**WW2_T**	**Walt Whitman**	**Drum Taps**	CM	MS	MS	MS	RK	ED
14	**WB1_T**	**William Blake**	**Songs of Innocence and Experience**	ED	ED	ED	ED	MS	ED
15	**WB2_T**	**William Blake**	**The Marriage of Heaven and Hell**	ED	EAP	ED	RF	EAP	ED
16	**STC_T**	**Samuel Taylor Coleridge**	**The Rime of the Ancient Mariner**	ED	RF	RF	RF	EAP	ED
17	**JL_T**	**Jack London**	**The Scarlet Plague**	RK	RK	RK	RK	RK	RF
18	**SBT_T**	**Samuel Butler**	**Erewhon**	HGW	EAP	CD	CD	HGW	ACD

The results presented in the table indicate that text-based similarity measures (such as the Centroid-Based Cosine Similarity approach) and linear discriminative models (LinearSVC and Logistic Regression) most consistently capture literary genre and stylistic similarities. For instance, the fact that Dorothy L. Sayers's *Whose Body?* is matched with Agatha Christie by all three "best-performing" methods is a notable example of accurate alignment, as both authors employ comparable narrative techniques within the detective fiction genre, particularly the "locked room" mystery,

logical deduction, and the use of period-specific British English. Similarly, the association of Mary Elizabeth Braddon's Victorian sensation novel *Lady Audley's Secret* with Charles Dickens appears plausible given their shared tendency to intertwine social criticism with melodrama and their use of extended, complex sentence structures.

Rudyard Kipling's recurring appearance as the "most similar author" across ten different test cases, including *Robinson Crusoe*, *Tom Sawyer Abroad*, *The Wind in the Willows*, and *Doctor Dolittle*, suggests that, despite differing cultural contexts, the colonial-adventure perspective and didactic tone toward animal characters contribute to common vector patterns. On the other hand, the unexpected pairing of Jonathan Swift's *Gulliver's Travels* with Christopher Marlowe via the Centroid-Based Cosine Similarity approach, or the inconsistent attribution of Samuel Butler's *Erewhon* to either H. G. Wells or Edgar Allan Poe, demonstrates that a classical-satirical writing style alone is insufficient for consistent genre-based classification. These inconsistencies, particularly in models such as Random Forest, Gaussian Naive Bayes, or K-Nearest Neighbors (KNN-3), suggest that factors such as stylistic variability and text length can distort results.

In summary, the Centroid-Based Cosine Similarity approach, LinearSVC, and Logistic Regression trio proves more effective in identifying real-world thematic, stylistic, and period-based similarities between literary works. In contrast, more erratic outcomes from Random Forest and especially GaussianNB and KNN-3 models highlight the need for cautious application when attempting to generalize to "unseen" authors.

For example, a reader who appreciates The Big Four by Agatha Christie is also likely to enjoy Whose Body? by Dorothy L. Sayers, as all three top-performing models consistently associate the two authors based on their shared narrative logic and genre conventions. Similarly, someone interested in Daniel Defoe's Robinson Crusoe might be inclined toward Jules Verne, given the consistent recommendation of Verne's adventure novels by the Centroid-Based Cosine Similarity approach, LinearSVC, and Logistic Regression, highlighting the overlap in exploratory themes and structured prose.

5.2.8.4. Author Prediction - Turkish (Unseen Authors)

Table 18. Turkish Author Prediction - Unseen Authors (Machine Learning and Cosine Similarity)

No	Tag	Author	Book	Cosine	LinearSVC	LogReg	RandomForest	GaussianNB	KNN-3
1	**SS_T**	**Samipaşazade Sezai**	**Sergüzeşt**	HRG	IP	HRG	SA	DC	SB
2	**YKK_T**	**Yakup Kadri Karaosmanoğlu**	**Kiralık Konak**	SA	SA	SA	HRG	SA	HRG
3	**RNGC_T**	**Reşat Nuri Güntekin**	**Çalıkuşu**	SA	SA	SA	SA	SA	SB
4	**CT_T**	**Canan Tan**	**Sızı**	NY	DC	DC	DC	DC	SB
5	**MO_T**	**Merve Özcan**	**Gözlerini Haramdan Sakın**	AU	AU	AU	DC	DC	SB
6	**SY_T**	**Sevinç Yavuz**	**Türk Seri Katiller**	AU	AU	AU	AU	DC	SB
7	**SFAH_T**	**Sait Faik Abasıyanık**	**Havuz Başı, Son Kuşlar**	IO	IO	IO	IO	ZL	SB
8	**TG_T**	**Tevfik Güran**	**19 Yüzyılda Osmanlı Ekonomisi Üzerine Araştırmalar**	HI	HI	HI	HI	HI	SB
9	**AC_T**	**Aydın Çetiner**	**21. Yüzyılda Savaş Stratejileri**	IO	HI	IO	IO	ZL	SB
10	**CCO_T**	**Cansu Canan Özgen**	**Türklerin Serüveni**	IO	IO	IO	IO	DC	SB

Table 18 (Continued). Turkish Author Prediction - Unseen Authors (Machine Learning and Cosine Similarity)

11	**EA_T**	**Erhan Afyoncu**	**Osmanlı'da Askeri İsyanlar ve Darbeler**	HI	HI	HI	IO	ZL	SB
12	**RE_T**	**Râşid Erer**	**Türklere Karşı Haçlı Seferleri**	HI	HI	HI	SA	ZL	SB
13	**ASI_T**	**Ahmet Şerif İzgören**	**Avucunuzdaki Kelebek**	DC	DC	DC	DC	DC	SB
14	**BS_T**	**Büşra Sanay**	**Yaşadım Demek İçin Ne Yapmalı**	DC	DC	DC	DC	DC	SB
15	**MSK_T**	**Mümin Sekman**	**Her Şey Seninle Başlar**	DC	DC	DC	DC	DC	SB
16	**ION_T**	**İpek Ongun**	**Mektup Arkadaşları**	SB	SB	SB	HRG	DC	SB
17	**SSA_T**	**Şevket Süreyya Aydemir**	**Suyu Arayan Adam**	SA	SA	SA	SA	SA	SB
18	**IOA_T**	**İhsan Oktay Anar**	**Puslu Kıtalar Atlası**	ES	ES	ES	SA	DC	SB
19	**NGK_T**	**Nurdan Gürbilek**	**Kör Ayna Kayıp Şark**	IG	IG	IG	HRG	DC	HRG
20	**NGKC_T**	**Nurdan Gürbilek**	**Kötü Çocuk Türk**	IG	IG	IG	DC	DC	SB
21	**PS_T**	**Peyami Safa**	**Yalnızız**	SA	SA	SA	HRG	SA	HRG
22	**RI3_T**	**Rıfat Ilgaz**	**Sosyal Kadınlar Partisi**	IO	IO	IO	IO	ZL	SB

The findings from the “unseen author” experiments conducted in this thesis demonstrate that content-based vector representations (Doc2Vec), when combined with the Centroid-Based Cosine Similarity approach, LinearSVC, and Logistic Regression models, can consistently capture similarities in literary genre and style.

Works by prominent authors of the Turkish Republican era, such as Yakup Kadri, Reşat Nuri, Peyami Safa, and Şevket Süreyya, frequently aligned with Sabahattin Ali across all three methods. Similarly, the clustering of personal development texts around Doğan Cüceloğlu and the consistent association of Ottoman historiography with the Halil İnalcık and İlber Ortaylı pair confirm that thematic and linguistic commonalities serve as strong discriminators within the vector space.

Within the detective fiction subset, Ahmet Ümit repeatedly emerged as the closest match to both contemporary thrillers (e.g., Merve Özcan) and true-crime narratives (e.g., Sevinç Yavuz), indicating that the model effectively captures the lexical features specific to crime writing. In contrast, Gaussian Naive Bayes's tendency to regress toward "average" centers such as Zülfü Livaneli (due to its assumption of unimodal distributions) and the over-association of KNN-3 with Samed Behrengi (likely resulting from its low k value) highlight the need for careful tuning of non-parametric models.

Moreover, mismatches such as the association of Sait Faik's short stories with İlber Ortaylı suggest that shared vocabulary between archaic Istanbul Turkish and historical writing may lead to contextual misclassifications. These cases underscore the necessity of marking domain-specific terms during text preprocessing and integrating genre-based filters within the recommendation system.

Overall, the results confirm that the combination of the Centroid-Based Cosine Similarity approach, LinearSVC, and Logistic Regression offers the highest consistency in both authorship attribution and content-based recommendation tasks. This also validates the practical applicability of the emerging recommendation engine, particularly when paired with strategies that prioritize readability and thematic coherence. For instance, a reader who enjoys Yakup Kadri Karaosmanoğlu's *Kiralık Konak* is likely to appreciate the works of Sabahattin Ali, as all top-performing models consistently group them based on their shared ideological depth and narrative tone. Likewise, those interested in Halil İnalcık's historical analyses are frequently directed toward İlber Ortaylı, confirming that the system successfully identifies authors with overlapping historiographical themes and stylistic rigor.

CHAPTER 6: CONCLUSION

This study demonstrated that literary texts can be effectively represented in a semantic vector space through the Doc2Vec algorithm and subsequently classified according to their genre and authorship using a combination of similarity metrics and machine learning classifiers. By applying the models to a curated corpus of Turkish and English books, including both seen and unseen authors, the research revealed consistent patterns of thematic and stylistic similarity that are in line with literary intuition and expert categorizations.

In genre classification, LinearSVC and Logistic Regression emerged as the most successful models for both English and Turkish datasets. On the English dataset with seen authors, both models achieved an accuracy of 93.1%, while on the Turkish dataset, LinearSVC reached a notably high accuracy of 92.0%. The Centroid-Based Cosine Similarity approach also performed competitively (89.7% for English, 80.0% for Turkish), highlighting the strength of document embeddings in capturing contextual proximity without parametric learning.

Out-of-sample evaluations reinforced the generalizability of the models. On unseen English authors, the Centroid-Based Cosine Similarity approach and Logistic Regression maintained excellent performance with 94.4% accuracy, correctly identifying literary genres such as children's literature, detective fiction, and poetry based on stylistic footprints alone. On unseen Turkish authors, although performance dropped slightly, as expected in a more linguistically and thematically diverse corpus, LinearSVC and Logistic Regression still achieved 77.3% accuracy, significantly outperforming other models. The poor performance of Random Forest (9.1%) and KNN-3 (4.5%) on this task emphasized the limitations of distance-based and ensemble methods when faced with sparse or uneven literary features.

In the authorship attribution task, results were equally promising. All three top models (the Centroid-Based Cosine Similarity approach, LinearSVC, and Logistic Regression) achieved 96.0% accuracy on the Turkish dataset and 93.1–96.6% on the English dataset. The consistency of these results, even across different linguistic systems and stylistic traditions, supports the hypothesis that authorial voice, characterized by word choice, syntax, and thematic structure, can be computationally

modeled and recognized. Moreover, out-of-sample experiments revealed that authors with distinctive narrative styles (e.g., Sabahattin Ali, Doğan Cüceloğlu, Halil İnalcık) could still be accurately approximated, often being matched with thematically or stylistically adjacent authors.

Importantly, this study also highlights the strengths and weaknesses of different modeling approaches. While the Centroid-Based Cosine Similarity approach provides a transparent and intuitive method for recommendation systems based on proximity in vector space, LinearSVC and Logistic Regression offer more discriminative power when subtle stylistic or thematic distinctions are needed. In contrast, Random Forest, Gaussian Naive Bayes, and KNN-3 models demonstrated inconsistent or noisy predictions, particularly in out-of-distribution scenarios, suggesting that their assumptions and sensitivity to data distribution make them less suitable for nuanced literary classification tasks.

In conclusion, the combination of document embeddings with linear classifiers represents a robust and scalable approach for genre classification, authorship attribution, and content-based recommendation in literary datasets. The results provide a solid foundation for future applications such as personalized reading recommendations, digital archiving, and literary style analysis. Future work could enhance these models with deeper linguistic features, genre-specific fine-tuning, and neural approaches to further capture semantic nuance and intertextuality in literature.

6.1. Suggestions for Future Studies

For future researchers interested in applying natural language processing techniques to literary corpora, several important recommendations can be made. First, the quality and balance of the dataset play a crucial role in achieving meaningful classification results. It is essential to ensure that each author and genre is adequately represented in both training and testing sets. Second, while traditional classifiers such as LinearSVC and Logistic Regression have proven effective, incorporating more advanced neural architectures, such as Transformer-based models, may provide deeper contextual understanding, especially for complex narrative styles. Third, careful preprocessing tailored to the domain (e.g., handling archaic language, non-standard orthography such

as spelling variation, or poetic forms) can significantly impact model performance. Moreover, researchers are encouraged to explore multi-task learning frameworks that combine genre and authorship prediction tasks, as well as cross-linguistic comparisons that can reveal universal stylistic markers across literary traditions. Finally, for real-world applications such as digital archives, educational tools and literary recommendation engines, integrating explainability and transparency into models will be vital for building trust and interpretability among non-technical users and scholars in the humanities.

REFERENCES

Ayanoğlu, E., Çolak, Z., Tanyel, T., Sarıoğlu, H.Y. and Diri, B. (2023) *Detection and classification of customer comments containing complaints*, European Journal of Science and Technology, Vol. 52, pp. 37–45.

Aydoğan, M. and Karcı, A. (2019) *Investigation of word similarities with word embedding methods*, Çukurova University Journal of the Faculty of Engineering and Architecture, Vol. 34 (2), pp. 181–195.

Baghirov, A. (2024) *Doğal dil işlemede makine öğrenme yöntemlerinin şair tanıma problemi üzerinde kıyaslanması*, Master's thesis, Ostim Technical University.

Bauersfeld, L., Romero, A., Muglikar, M. and Scaramuzza, D. (2023) *Cracking double-blind review: authorship attribution with deep learning*, PLOS ONE, Vol. 18 (6), e0287611.

Bilgin, M. and Şentürk, İ.F. (2017) *Sentiment analysis on Twitter data with semi-supervised Doc2Vec*, *Proceedings of the International Conference on Computer Science and Engineering (UBMK)*, Antalya, Turkey, 5–8 October. IEEE, pp. 661–666.

Budiarto, A., Rahutomo, R., Putra, H.N., Cenggoro, T.W., Kacamarga, M.F. and Pardamean, B. (2021) *Unsupervised news topic modelling with Doc2Vec and spherical clustering*, Procedia Computer Science, Vol. 179, pp. 40–46.

Cerqueira, W.R.P., Oliveira, A.F.M., Pereira, A.F., Ferreira, R.A.C. and de Carvalho, A.C.P.L.F. (2023) *Morphofunctional variations of the Echinometra lucunter (Echinoidea) on Bahia Coast, Brazil*, Academia Biology, Vol. 1(3).

Çelenli, H.İ., Öztürk, S.T., Şahin, G., Gerek, A. and Ganiz, M.C. (2018) *Document embedding based supervised methods for Turkish text classification*, *Proceedings of the 3rd International Conference on Computer Science and Engineering (UBMK)*, Sarajevo, Bosnia and Herzegovina, 20–23 September. IEEE, pp. 477–482.

da Rocha Bartolomei, B. and Drummond, I.N. (2020) *Authorship attribution of Brazilian literary texts through machine learning techniques*, in Cerri, R. and Prati, R.C. (eds) *Intelligent Systems. BRACIS 2020* (Lecture Notes in Computer Science, vol. 12319). Cham: Springer, pp. 389–402.

Demir, B. (2024) *Prediction of psychological distress in persons through narrative writing by using natural language processing (NLP)*, PhD thesis, Dokuz Eylül University.

Deniz, E., Erbay, H. and Coşar, M. (2019) *Classification of Turkish e-mails with Doc2Vec*, *Proceedings of the 1st International Informatics and Software Engineering Conference (UBMYK)*, Ankara, Turkey. IEEE, pp. 1–4.

Doğan, İ. (2023) *Doğal dil işleme ve makine öğrenmesi yöntemleri kullanılarak stilometri analizi ile Türkçe metinlerde eser-yazar eşleştirme*, Master's thesis, Konya Technical University.

Doğru, H.B., Tilki, S., Jamil, A. and Hameed, A.A. (2021) *Deep learning-based classification of news texts using Doc2Vec model*, *Proceedings of the 1st International Conference on Artificial Intelligence and Data Analytics (CAIDA)*, Riyadh, Saudi Arabia. IEEE, pp. 91–96.

Erdinç, H.Y. and Güran, A. (2019) *Semi-supervised Turkish text categorization with Word2Vec, Doc2Vec and FastText algorithms*, *Proceedings of the 27th Signal Processing and Communications Applications Conference (SIU)*, Sivas, Turkey. IEEE, pp. 1–4.

Fedotova, A., Romanov, A., Kurtukova, A. and Shelupanov, A. (2022) *Authorship attribution of social media and literary Russian-language texts using machine learning methods and feature selection*, Future Internet, Vol. 14 (1), Article 4 [Online]. Available at: https://doi.org/10.3390/fi14010004 (Accessed: 20 May 2025).

Goyal, A. and Prem Prakash, V. (2022) *Statistical and deep learning approaches for literary genre classification*, in Tiwari, S. et al. (eds) *Advances in Data and Information Sciences,* Lecture Notes in Networks and Systems, Vol. 318, Singapore: Springer, pp. 297–305.

Gupta, S., Agarwal, M. and Jain, S. (2019) *Automated genre classification of books using machine learning and natural language processing*, *Proceedings of the 9th International Conference on Cloud Computing, Data Science & Engineering (Confluence)*, Noida, India, 10–11 January. IEEE, pp. 269–272.

Hu, X., Wang, Y. and Wu, Q. (2014) *Multiple authors detection: a quantitative analysis of Dream of the Red Chamber*, Advances in Adaptive Data Analysis, Vol. 6 (4), Article 1450012.

Kaya, S. (2018) *Doğal dil işleme teknikleriyle yazar-kitap tanıma*, Master's thesis, Istanbul Aydın University.

Kınık, D. (2020) *TF-IDF ve Doc2Vec tabanlı metin sınıflandırma sisteminin başarı değerinin ardışık kelime grubu tespiti ile arttırılması*, Master's thesis, Doğuş University.

Kınık, D. and Güran, A. (2021) Enhancing the performance of TF-IDF and Doc2Vec based Turkish text categorization system with phrase modeling, European Journal of Science and Technology, Vol. 21, pp. 323–332.

Korkmaz, S. (2021) *Makine öğrenme yöntemleri kullanılarak doğal dil işleme tabanlı şair tanıma*, Master's thesis, Erciyes University.

Kuzman, T. and Ljubešić, N. (2025) *Automatic genre identification: a survey*, Language Resources and Evaluation, Vol. 59 (1), pp. 537–570.

Le, Q.V. and Mikolov, T. (2014) *Distributed representations of sentences and documents*, arXiv [Online]. Available at: https://arxiv.org/abs/1405.4053 (Accessed: 20 May 2025).

McCarthy, J., Minsky, M.L., Rochester, N. and Shannon, C.E. (2006) *A proposal for the Dartmouth summer research project on artificial intelligence, 31 August 1955*, AI Magazine, Vol. 27 (4), p. 12.

Merriam-Webster (2024) Genre. *Merriam-Webster Dictionary* [Online]. Available at: https://www.merriam-webster.com/dictionary/genre (Accessed: 20 May 2025).

Mikolov, T., Chen, K., Corrado, G. and Dean, J. (2013) *Efficient estimation of word representations in vector space*, arXiv [Online]. Available at: https://arxiv.org/abs/1301.3781 (Accessed: 20 May 2025).

Monte-Serrat, D.M., Machado, M.T. and Ruiz, E.E.S. (2021) *A machine learning approach to literary genre classification on Portuguese texts: circumventing NLP's standard varieties*, *Proceedings of the 13th Brazilian Symposium in Information and Human Language Technology (STIL 2021)*, Porto Alegre, Brazil. Sociedade Brasileira de Computação, pp. 255–264.

Mumcuoğlu, E. (2022) *Prediction of outcomes in higher courts of Turkey using natural language processing*, Master's thesis, Bilkent University.

Sakal, T. and Proulx, S. (2023) *Revisiting the relevance of traditional genres: a network analysis of fiction readers' preferences*, arXiv [Online]. Available at: https://doi.org/10.48550/arXiv.2303.05080 (Accessed: 20 May 2025).

Seyfioğlu, S.M. and Demirezen, M.U. (2017) *A hierarchical approach for sentiment analysis and categorization of Turkish written customer relationship management data*, *Proceedings of the Federated Conference on Computer Science and Information Systems (FedCSIS)*, Vol. 11, pp. 361–365.

Shavrina, T. (2020) *Genre classification on text-internal features: a corpus study*, *ARANEA 2018 – Web Corpora as a Language Training Tool*, Comenius University, Bratislava, 23–24 November 2018, pp. 134–147 [Online]. Available at: https://www.academia.edu/71481286/Genre_Classification_on_Text_Internal_Features_a_Corpus_Study (Accessed: 20 May 2025).

Stamatatos, E., Fakotakis, N. and Kokkinakis, G. (2000) *Automatic text categorization in terms of genre and author*, Computational Linguistics, Vol. 26 (4), pp. 471–495.

Torun, H. and İnner, A.B. (2022) *A method for similarity detection in vector space by summarizing news articles*, *Proceedings of the 30th Signal Processing and Communications Applications Conference (SIU)*, Safranbolu, Turkey. IEEE, pp. 1–4.

Vakkari, P. and Mikkonen, A. (2020) *The role of readers' literary preferences in predicting success in fiction search*, Journal of Documentation, Vol. 76 (1), pp. 317–332.

Walkowiak, T. (2023) *Authorship attribution of literary texts using named entity masking and MaxLogit-based sequence classification for varying text lengths*, in Rutkowski, L. et al. (eds) *Artificial Intelligence and Soft Computing* (Lecture Notes in Computer Science, vol. 14125). Cham: Springer, pp. 296–303.

Yegen, T. (2023) *Türkçe doğal dil işleme araçlarının ilk ve ortaokul fen bilimleri ders kitapları analizinde kullanımı: sözcük ve sözcük kökü frekanslarının karşılaştırmalı analizi*, Master's thesis, Niğde Ömer Halisdemir University.

Yönet, E. (2023) *Şiir kategorisinin doğal dil işleme yöntemleri kullanılarak tahmin edilmesi*, Master's thesis, Konya Technical University.