-- Biannual Scientific of Journal Language Science -----
Spring & Summer 2025, 12(21), 45-76
ls.atu.ac.ir
DOI: 10.22054/ls.2021.60784.1453

ATU PRESS

Original Research

eISSN: 2538-2551 ISSN: 2423-7728 Received:08/May/2021 * Revised:22/Aug/2021 * Accepted: 08/Sep/2021

Comparing Basic ML Algorithms for Classifying Allusive vs. Non-Allusive Persian Poetry

Parisa Mohammadian Kalkhoran — M.Sc. in Computational Linguistics, Sharif University of Technology, Tehran, Iran

Mohammad Bahrani * — Assistant Professor, Department of Computer, Faculty of Statistics Mathematics and Computer, Allameh Tabataba'i University, Tehran, Iran

Abstract

The present study aims to evaluate the performance of several machine learning methods in classifying Persian poetry into two categories: verses containing allusion and those without allusion. To this end, the following supervised learning algorithms were applied: Naive Bayes, Support Vector Machine (SVM), Decision Tree, Random Forest, k-Nearest Neighbors (k-NN), Logistic Regression, and Multilayer Perceptron (MLP). Labeled data were collected and stored in two text files. Each verse was then transformed into a numerical vector representation. After merging the datasets and splitting them into training and testing subsets, each algorithm was implemented on the training data and evaluated on the test set to assess its classification accuracy. The output of each algorithm consisted of predicted labels assigned to the verses. The evaluation methodology used was Leave-One-Out Cross-Validation (LOOCV). The results indicate that Naive Bayes (76.09%), Logistic Regression (76.09%), Multilayer Perceptron (75.22%), and Support Vector Machine (74.35%) performed better than the other algorithms in terms of

* Corresponding Author: bahrani@atu.ac.ir

How to Cite: Mohammadian Kalkhoran, P., & Bahrani, M. (2025). Comparing Basic ML Algorithms for Classifying Allusive vs. Non-Allusive Persian Poetry. *Language Science*, *12*(21), 45-76. doi: 10.22054/ls.2021.60784.1453.

accuracy. Overall, considering additional metrics such as F1-score and execution time, Naive Bayes demonstrated the best performance among the tested methods.

1. Introduction

Persian poetry is a rich cultural heritage filled with literary devices, one of which is allusion, a rhetorical technique that enriches meaning through indirect references to well-known stories, sayings, or cultural elements. As society becomes increasingly intertwined with technology, there is a growing need to leverage machine learning to preserve and study such literary elements. This study explores the application of supervised machine learning algorithms to classify Persian couplets as either containing allusion or not, thus contributing to the intersection of literary analysis and artificial intelligence.

2. Literature Review

Allusion has long been recognized as a central rhetorical device in Persian literature. Classical scholars such as Qeys Razi (1372: 279) and the author of Anvār al-Balāgha (17th century AD) defined it as a brief, implicit reference to the well-known texts, events, or sayings. Shamisa (1381: 22) expanded the definition, incorporating references to folklore, customs, and scientific concepts as part of the broader allusive framework. These definitions consistently highlight conciseness, implicitness, and the need for cultural familiarity to interpret the text accurately.

A large number of literary studies have focused on identifying allusions in the works of specific poets using manual and qualitative research methods. The field of computational literary studies is relatively underexplored in Persian, though some efforts have been made. For instance, works by Majiri and Minaei (2008) applied text mining techniques for prosodic analysis, while others such as Azin & Bahrani (2014) and Javanmardi & Akbari (2017) explored authorial style classification.

Machine learning, particularly supervised learning, has recently emerged as a promising approach in text classification, including applications in authorship attribution, genre detection, and sentiment

analysis. However, the use of ML for detecting literary devices like allusion remains limited, especially in Persian literary texts. Two general approaches are used for text classification: vocabulary-based (dictionary-driven) and machine learning-based. This study focuses on the latter, aiming to automatically classify verses using statistical and linguistic features rather than predefined word lists.

3. Methodology

This study followed a seven-step machine learning pipeline using Python programming language:

Data Collection: Allusive verses were extracted from *Farhang-e Asatir va Dastanvareha* by Mohammad Jafar Yahaghi (1388) and were typed manually. Non-allusive verses were collected from the Ganjoor website (ganjoor.net). The dataset included 300 allusive and 160 non-allusive couplets, stored in separate *.txt* files.

Data Preprocessing: Text normalization and cleaning involved punctuation removal, spacing corrections, and standardization. Tokenization and stop-word removal were conducted using a curated stop-word list tailored to the dataset. Due to the limited data size and poetic language variability, lemmatization was omitted to preserve original word forms.

Feature Extraction and Vectorization: The remaining 2107 unique words after preprocessing were treated as features. Two vectorization techniques were used:

TF-IDF (Term Frequency–Inverse Document Frequency)

Binary Encoding (presence/absence of words)

The effect of the vectorization method on classification performance was observed, especially in Logistic Regression and MLP classifiers.

1. Dataset Splitting: The dataset was divided into training and test sets using Leave-One-Out Cross-Validation (LOOCV) to maximize accuracy evaluation on a small dataset.

2. Model Training: Seven supervised machine learning algorithms were trained: Naive Bayes, Support Vector Machine (SVM), Decision Tree, Random Forest, K-Nearest Neighbors (KNN), Logistic Regression, Multilayer Perceptron (MLP).
3. Model Evaluation: Accuracy, F1-score, and execution time were used as performance metrics. LOOCV ensured robust evaluation by testing each data point independently.
4. Output Generation: The predicted label for each couplet was obtained using the trained models. The predicted labels were compared with actual labels to compute accuracy.

4. Results

The accuracy scores for the best-performing models were: Naive Bayes: 76.09%, Logistic Regression: 76.09%, Multi-layer Perceptron: 75.22%, SVM: 74.35%.

Other models like Decision Tree, Random Forest, and KNN showed lower performance. Notably, most models showed negligible performance differences between the two vectorization methods, except for Logistic Regression and MLP, which exhibited a 10% accuracy gap.

5. Discussion

The results suggest that simpler models like Naive Bayes can perform competitively in literary text classification tasks, especially when the dataset is limited. The choice of vectorization method had minimal impact on overall accuracy, underscoring the importance of careful feature selection over sophisticated encoding. Despite the challenges of working with poetic texts—including varied structure and archaic language—machine learning offers promising results in detecting abstract literary features like allusion.

6. Conclusion

This study aimed to evaluate the performance of several supervised machine learning algorithms in classifying Persian poetry into two categories: verses containing allusions and those without. Algorithms

such as Naive Bayes, Support Vector Machine, Decision Tree, Random Forest, k-Nearest Neighbors, Logistic Regression, and Multilayer Perceptron were applied, with their performance assessed using Leave-One-Out Cross-Validation (LOOCV).

The research highlighted several challenges in automatic text classification, especially in Persian poetry, including high feature dimensionality, lexical ambiguities, flexible sentence structures, and stylistic diversity among poets. Specifically, the identification of literary allusions proved to be complex due to the implicit nature of many references, variation in allusion types (e.g., Quranic, mythological, national), and the evolving nature of allusive language over time.

Despite these difficulties, some algorithms demonstrated promising results, suggesting that with larger, more diverse datasets and the use of advanced techniques such as transfer learning and pre-trained language models, future efforts in this area could achieve higher accuracy and robustness. The intersection of computational methods and literary analysis presents a valuable avenue for deeper understanding and automated processing of Persian literary texts.

Keywords: allusion, Persian poetry, text classification, machine learning, Natural Language Processing.

-- دوفصلنامه علمی علم زبان ---------
دوره ۱۲، شماره ۲۱، بهار و تابستان ۱٤۰٤، ٤٥ـ٧٦
ls.atu.ac.ir
DOI: 10.22054/ls.2021.60784.1453

مقاله پژوهشی
تاریخ دریافت: ۱۴۰۰/۰۲/۱۸ * تاریخ بازنگری: ۱۴۰۰/۰۶/۰۱ * تاریخ پذیرش: ۱۴۰۰/۰۶/۱۸
ISSN: ۲۴۲۳-۷۷۲۸
eISSN: ۲۵۳۸-۲۵۵۱

مقایسهٔ عملکرد الگوریتم‌های پایهٔ یادگیری ماشین در دسته‌بندی اشعار فارسی به دو گروه تلمیح‌دار و بدون‌تلمیح

پریسا محمدیان کلخوران — دانش‌آموختهٔ کارشناسی ارشد زبان‌شناسی رایانشی، دانشگاه صنعتی شریف، تهران، ایران

محمد بحرانی * — استادیار، گروه رایانه، دانشگاه علامه طباطبائی، تهران، ایران

چکیده

هدف از پژوهش حاضر بررسی عملکرد چند روش یادگیری ماشین در دسته‌بندی اشعار فارسی به دو گروه تلمیح‌دار و بدون‌تلمیح است. به‌این‌منظور، از روش‌های نظارت‌شدهٔ بیز ساده، ماشین بردار پشتیبان، درخت تصمیم، جنگل تصادفی، k نزدیک‌ترین همسایه، رگرسیون لجستیک و الگوریتم پرسپترون چندلایه استفاده شد. پس از جمع‌آوری داده‌های برچسب‌خورده در قالب دو فایل متنی، هرکدام از ابیات به بردار عددی تبدیل شدند. پس از ادغام داده‌ها و تقسیم آنها به دو دستهٔ آموزش و آزمون، الگوریتم مدنظر بر روی داده‌های آموزشی پیاده‌سازی و بر روی داده‌های آزمون، آزمایش گردید تا دقت عملکرد الگوریتم سنجیده شود. خروجی هر الگوریتم، برچسب پیش‌بینی‌شده توسط ماشین برای ابیات موردنظر بود و برای ارزیابی الگوریتم‌ها از روش LOOCV استفاده شد. نتایج ارزیابی نشان داد که الگوریتم‌های بیز ساده ۷۶/۰۹٪، رگرسیون لجستیک ۷۶/۰۹٪، پرسپترون چند لایه ۷۵/۲۲٪ و ماشین بردار پشتیبان ۷۴/۳۵٪ نسبت به الگوریتم‌های دیگر عملکرد بهتری دارند. درمجموع و با توجه به سایر معیارها، از جمله معیار اف ـ ۱ و زمان اجرا، می‌توان گفت که بهترین عملکرد مربوط به الگوریتم بیز ساده بود.

کلیدواژه‌ها: تلمیح، شعر فارسی، دسته‌بندی متن، یادگیری ماشین، پردازش زبان طبیعی.

* نویسنده مسئول: bahrani@atu.ac.ir

۱. مقدمه

«شعر جوهر حیات فرهنگی اقوام است». شفیعی کدکنی

شاعران بزرگ فارسی همواره در گذر زمان کوشیده‌اند تا به شیوه‌های مختلف و با استفاده از انواع هنرهای زبانی، عالی‌ترین مفاهیم انسانی را با اشاره و استعاره، به مردم زمان خود و حتی مردم نسل‌های بعد، منتقل کنند. یکی از شگردهایی که شاعر با استفاده از آن می‌تواند ظرفیت معنایی کلام را افزایش دهد و آن را ماندگار کند، هنر تلمیح[1] است.

کزازی (۱۳۷۲: ۱۱) در زمینهٔ این هنر زبانی چنین می‌گوید که «چشمزد یا تلمیح، آرایه‌ای است درونی که سخنور بدان، سخت کوتاه، از داستانی، دَستانی (= مثل)، گفته‌ای و هرچه از این گونه، سخن در میان آورد و آن داستان، یا دستان، یا گفته را به یکبارگی در ذهن سخن‌دوست برمی‌انگیزد. چشمزد آرایه‌ای است که سخنور به یاری آن می‌تواند بافت معنایی سروده را نیک ژرفا و گرانمایگی ببخشد و دریایی از اندیشه‌ها را در کوزه‌ای تنگ از واژگان فرو ریزد. از آن جاست که سخنورانی پندارآفرین و اندیشه‌انگیز، چون حافظ که سروده‌های خویش را با مایه‌ها و لایه‌هایی از نگاره‌ها و انگاره‌های شاعرانه، پی‌درپی و تودرتوی، ژرفی و شگرفی می‌دهند و از این آرایه بسیار بهره برده‌اند».

شمیسا (۱۳۸۱: ۲۲) نیز، در این زمینه معتقد است که «تلمیح از جمله صنایع معنوی در علم بدیع است و بدیع علمی است که پیرامون جنبه‌های تحسین و نیکویی کلام بحث می‌کند. به‌عبارت‌دیگر، بدیع مجموعهٔ شگردها و فنونی است که کلام عادی را کم‌وبیش به کلام ادبی تبدیل می‌کند و یا کلام ادبی را به سطح والاتری از ادبی‌بودن می‌رساند و در بین اجزای کلام، تناسب و رابطه‌های خاصی برقرار می‌کند».

هر شعر یادگار فرهنگ و تمدنی است و مخاطب باید لایه‌های تودرتوی شعر را طی کند تا بتواند به درکی مشترک با شاعر نائل شود، در هوای آن فرهنگ و تمدن نفس بکشد و در نهایت، به پشتوانه‌های عمیق فکری کلام دست پیدا کند. شعرا زبان را از

---

1. allusion

سطح عادی و روزمرهٔ آن، به سطحی والا و هنری ارتقا داده‌اند و اینک، این میراث ارزشمند در دستان ماست و در انتظار شنونده‌ای مشتاق که با گوش جان آن را بشنود و در حفظش بکوشد. این شنونده، اگر با اشارات شاعر بیگانه باشد، بی‌تردید مصالح لازم برای ساخت بنای فکری را نخواهد داشت.

شرایط زندگی امروز و آمیختگی آن با فناوری و نزدیکی روزافزون ماشین با انسان ایجاب می‌کند که به‌جای قائل‌شدن به استقلال شعر و ادبیات و تلقی هوش مصنوعی به‌منزلهٔ دشمن تفکر و انسانیت و سایر جنبه‌های زیبای انسانی، از این دستاورد عظیم بشری با هدف آشنایی بیشتر نسل جدید با فرهنگی که دستاورد بزرگان روزگار است، بهره ببریم. در وضعیت کنونی، دست یازیدن به این هدف شاید بیشتر به آرزو شبیه باشد، اما غیرممکن نیست. ازاین‌رو، پژوهش حاضر قصد دارد در مسیر تحقق این هدف، قدمی کوچک بردارد. به‌این‌منظور، حوزهٔ شعر و از این حوزه هم، صنعت ادبی تلمیح انتخاب شده است، چراکه چنانکه محبتی (۱۳۸۰: ۱۱۳) نیز به آن باور دارد، «مهم‌ترین آثار ادبی جهان، بیشترین بار معنایی خود را بر دوش تلمیح نهاده‌اند، چون از یک‌سو مانند تضمین[۱] مستقیماً از گفته‌ها و گفتارهای دیگران سود نمی‌جوید و درعین‌حال، تاریخ فرهنگی و بار معنایی گفتارهای گذشته را به درون خود می‌کشد».

هدف اصلی پژوهش حاضر، استفاده از الگوریتم‌های یادگیری ماشین[۲]، از جمله بیز ساده[۳]، K ـ نزدیک‌ترین همسایه[۴]، درخت تصمیم[۵]، جنگل تصادفی[۶]، رگرسیون لجستیک[۷]، ماشین بردار پشتیبان[۸] و پرسپترون چند لایه[۹]، در تفکیک اشعار تلمیح‌دار از

---

۱. تضمین از آرایه‌های ادبی و به معنی آوردن آیه، حدیث، مثل، سخنی مشهور و یا مصراع یا بیتی از شاعری دیگر در بین سخن خود است.

2. machine learning algorithms
3. naive bayes
4. k-nearest neighbors (KNN)
5. decision tree
6. random forest
7. logistic regression
8. support vector machine (SVM)
9. multilayer perceptron

بدون‌تلمیح و مقایسهٔ آنهاست. لازم به ذکر است که در این پژوهش، هر بیت به‌عنوان یک سند متنی[1] در نظر گرفته شده است.

۲. پیشینهٔ پژوهش

به باور شمس‌الدین محمد قیس رازی (۱۳۷۲: ۲۷۹)، نویسندهٔ بزرگ قرن ششم هجری، تلمیح «آن است که الفاظ اندک، بر معانی بسیار دلالت کند و لمح، جستن برق باشد و لمحه یک نظر بود و چون شاعر چنان سازد که الفاظ اندک بر معانی بسیار دلالت کند، آن را تلمیح گویند».

صاحب انوارالبلاغه (سدهٔ یازدهم ه.ق) نیز دربارهٔ تلمیح می‌گوید که «آن است که از فحوای کلام اشاره شود به شعری یا قصه‌ای یا مثل مشهوری بدون آن که آن شعر یا قصه یا مثل مذکور شوند، خواه آن کلام نظم بوده باشد یا نثر» (مازندرانی، ۱۳۷۵).

داد، مؤلف فرهنگ اصطلاحات ادبی (۱۳۸۵)، به معنای لغوی تلمیح (به گوشهٔ چشم اشاره کردن، نگاه و نظر کردن) اشاره می‌کند و پس از تعریف تلمیح در اصطلاح بدیع، اقسام تلمیح از نگاه منتقدان غربی را اینگونه برمی‌شمارد: اشاره به اشخاص و حوادث، اشاره به زندگی شخصی یا نام خود گوینده (و این همان است که در بلاغت فارسی استشهاد نامیده می‌شود)، تلمیح استعاری و تلمیح تقلیدی.

با توجه به تعاریف متعددی که از تلمیح در دوره‌های مختلف دیده می‌شود، می‌توان گفت که در تلمیح این عناصر مشترک است: ایجاز (اشارهٔ کوتاه در حد گوشهٔ چشم) و غیرصریح‌بودن، به‌گونه‌ای که اگر شاعر کل داستان یا شعر مدنظرش را به‌طور کامل شرح دهد، دیگر تلمیح محسوب نمی‌شود.

البته اختلافات اندکی هم در تعریف تلمیح، میان ادبا وجود دارد؛ برای مثال چنانکه در دایره‌المعارف اسلامی آمده است، برخی با مقدم‌داشتن میم بر لام، این واژه را تملیح، از ریشهٔ ملح به معنای نمکِ به‌اندازه در طعام‌کردن و نیز، نمکین‌کردن کلام با استفاده

1. document

از پاره‌ای اشارات، حکایات و وقایع مشهور دانسته‌اند، اما گروهی از علمای علم بدیع، صورت اخیر را خطا و تلمیح را صواب دانسته‌اند.

شمیسا (۱۳۷۸) نیز، در فرهنگ تلمیحات خود، معنای تلمیح را گسترش می‌دهد و اشاره به فرهنگ عامه و عقاید و آداب و رسوم و علوم قدیم را هم بخشی از تلمیح قلمداد می‌کند. وی اذعان دارد که از معنای تلمیح در کتب سنتی پا فراتر می‌گذارد و گاهی به مواردی اشاره می‌کند که حاوی اشاره به داستانی نیست، بلکه اشاره به مطلبی دارد که آگاهی‌نداشتن از آن، درک بیت را غیرممکن می‌سازد.

در میان پژوهش‌های ادبی، آثار نسبتاً زیادی در حوزهٔ تلمیح به چشم می‌خورد، به‌ویژه در زمینهٔ بررسی موردی کاربرد تلمیح در آثار شاعران، چه اشعار کهن و چه اشعار مربوط به دوران معاصر و حتی شعر نو.

از اوایل دههٔ ۹۰ قرن بیستم شاهد ظهور رویکرد یادگیری ماشین به‌منظور دسته‌بندی متون[1] هستیم، اما این رویکرد تاکنون در حوزهٔ ادبیات زبان فارسی، ورود قابل‌توجهی نداشته است. بیشتر پژوهش‌های پیشین در این حوزه به شناسایی سبک و ویژگی‌های زبانی مؤلف در آثار ادبی پرداخته‌اند. از جمله این پژوهش‌ها، مقالهٔ مجیری و مینایی (۱۳۸۷) است که به کاربرد متن‌کاوی در تشخیص وزن عروضی اشعار فارسی می‌پردازد و یا مقالهٔ آذین و بحرانی (۱۳۹۳) و جوانمردی و اکبری (۱۳۹۷) است که در آنها هدف از پژوهش، شناسایی سبک شاعر و دسته‌بندی و تفکیک اشعار شاعران است. تیزهوش[2] (2008) هم جداسازی شعر از متن عادی در محیط وب را موضوع پژوهش خود قرار داده است.

در حالت کلی، برای تشخیص تلمیح در اشعار می‌توان دو رویکرد اصلی متصور شد: رویکرد یادگیری ماشین و رویکرد مبتنی بر واژگان[3]. رویکرد یادگیری ماشین از الگوریتم‌های معروف یادگیری که مبتنی بر ویژگی‌های زبانی هستند، استفاده می‌کند، در صورتی که رویکرد مبتنی بر واژگان، بر اساس واژه‌نامه‌ای بنا می‌شود که مجموعه‌ای

1. text classification
2. Tizhoosh, H. R.
3. lexicon-based approach

از لغات و واژه‌های ازپیش‌تعیین‌شده است. در پژوهش حاضر به رویکرد اول توجه شده است.

همانند داده‌کاوی[1] که جستجویی برای کشف الگو در داده‌هاست، متن‌کاوی[2] نیز، جستجو در متون برای کشف الگوی آنهاست. دسته‌بندی متن یکی از زیرشاخه‌های اصلی متن‌کاوی است؛ علمی که بر اساس داده‌های قبلی که دارای برچسب[3] هستند، مدلی برای پیش‌بینی برچسب داده‌های جدید می‌سازد. هدف از این روش، یادگیری تابعی[4] است که الگوهای (بردارهای ویژگی) ورودی را به برچسب‌های متناظرشان نگاشت کند. در واقع، این فرایند در مرحلهٔ آموزش[5] انجام می‌گیرد. بدیهی است که برچسب‌های واقعی الگوهای آموزشی، از پیش مشخص شده‌اند.

در مرحلهٔ آزمون[6]، الگوهایی که برچسب آنها مشخص نیست، به سامانه داده می‌شوند و سامانهٔ طراحی‌شده، به کمک تابع خود، خروجی یا برچسب آنها را پیش‌بینی می‌کند. به‌بیان‌دیگر؛ دسته‌بندی به فرایند قراردادن نمونه‌های جدید در دسته‌های مختلف بر اساس داده‌های قدیمی اشاره دارد و به‌این‌منظور، به یک مدل یا الگوریتم دسته‌بند نیاز است.

الگوریتم‌های طبقه‌بندی مشهور عبارت‌اند از: شبکه‌های عصبی[7]، انواع درخت تصمیم (CART، C4.5، J48، ID3 و ...)، K ـ نزدیک‌ترین همسایه، ماشین بردار پشتیبان، رگرسیون لجستیک، دسته‌بند بیز ساده و غیره که انتخاب هرکدام از آنها می‌تواند به موارد زیر وابسته باشد:

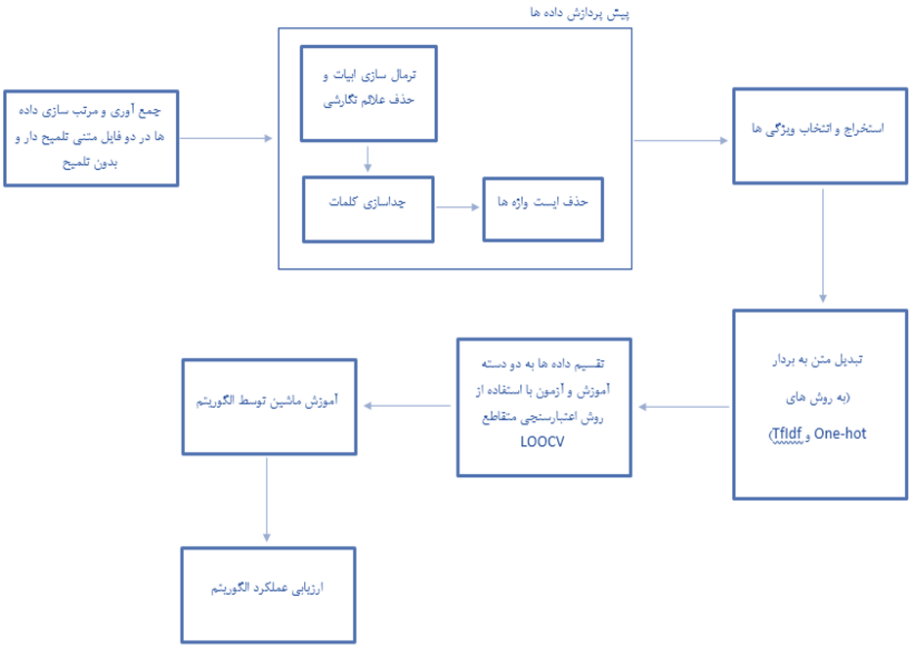
- ساختار و نحوهٔ توزیع داده‌ها
- تعداد طبقه‌ها
- پیچیدگی زمانی ساخت مدل

1. data mining
2. text mining
3. label
4. function
5. train
6. test
7. neural networks

۳. روش

در این بخش، به روش اجرای پژوهش در ۷ مرحله پرداخته شده است که عبارت‌اند از: جمع‌آوری و مرتب‌سازی داده‌ها[1]، پیش‌پردازش داده‌ها[2]، استخراج و انتخاب ویژگی‌ها[3]، تبدیل متن به بردار[4]، تقسیم داده‌ها به دو دستهٔ آموزش و آزمون، آموزش مدل‌های یادگیری ماشین و ارزیابی عملکرد مدل‌ها. مراحل پیاده‌سازی[5] در شکل ۱ نشان داده شده است. لازم به ذکر است که در تمام این مراحل، از زبان برنامه‌نویسی پایتون[6] استفاده شده است.

شکل ۱. مراحل پیاده‌سازی



1. data collection
2. preprocessing
3. feature extraction
4. text vectorization
5. implementation
6. Python

در گام اول، برای جمع‌آوری داده‌های دستهٔ تلمیح‌دار، از کتاب فرهنگ اساطیر و داستان‌واره‌ها در ادبیات فارسی (۱۳۸۸)، اثر یاحقی استفاده شد و ابیات آن به‌صورت دستی تایپ گردید. اشعار بدون‌تلمیح هم از سایت گنجور[1] گردآوری شد. در کل، بیشتر داده‌ها از اشعار حافظ انتخاب شده‌اند. نتایج حاصل از گردآوری داده‌ها، شامل ۳۰۰ بیت تلمیح‌دار و ۱۶۰ بیت بدون‌تلمیح، در دو فایل متنی مجزا با فرمت txt ذخیره شدند، به گونه‌ای که هر بیت نشان‌دهندهٔ یک سند متنی جداگانه است. نمونه‌هایی از داده‌ها در جدول ۱ آورده شده‌اند.

جدول ۱. نمونه‌هایی از داده‌های آموزشی

ابیات تلمیح‌دار	ابیات بدون‌تلمیح
گرت هواست که با خضر همنشین باشی / نهان ز چشم سکندر، چو آب حیوان باش	همه کارم ز خودکامی، به بدنامی کشید آخر / نهان کی ماند آن رازی، کز او سازند محفل‌ها
مقام عیش میسر نمی‌شود بی‌رنج / بلی به حکم بلا بسته‌اند عهد الست	ز عشق ناتمام ما جمال یار مستغنی است / به آب و رنگ و خال و خط چه حاجت روی زیبا را
من آن آیینه را روزی به دست آرم سکندروار / اگر می‌گیرد این آتش زمانی ور نمی‌گیرد	غرور حسنت اجازت مگر نداد ای گل / که پرسشی نکنی عندلیب شیدا را
زبان مور به آصف دراز گشت و رواست / که خواجه خاتم جم یاوه کرد و باز نجست	دل می‌رود ز دستم صاحب‌دلان خدا را / دردا که راز پنهان خواهد شد آشکارا
من ملک بودم و فردوس برین جایم بود / آدم آورد درین دیر خراب آبادم	ای پیک راستان خبر یار ما بگو / احوال گل به بلبل دستان سرا بگو
دوش وقت سحر از غصه نجاتم دادند / وندر آن ظلمت شب آب حیاتم دادند	ناصحم گفت که جز غم چه هنر دارد عشق / برو ای خواجه عاقل هنری بهتر از این
باز ارچه گاه‌گاهی، بر سر نهد کلاهی / مرغان قاف دانند آیین پادشاهی	ساقی و مطرب و می، جمله مهیاست ولی / عیش بی یار مهیا نشود، یار کجاست
جهان پیر است و بی‌بنیاد، ازین فرهادکش فریاد / که کرد افسون و نیرنگش ملول از جان شیرینم	گفتم که کی ببخشی بر جان ناتوانم / گفت آن زمان که نبود جان در میانه حائل

1. ganjoor.net

مرحلهٔ پیش‌پردازش داده‌ها در اغلب موارد، از جمله مهم‌ترین و زمان‌برترین گام‌های پروژه‌های داده‌کاوی است که انجام درست آن به بالارفتن دقت مدل می‌انجامد. پیش‌پردازش متن عمدتاً در آغاز فرایند متن‌کاوی و با توجه به یکی از اهداف زیر انجام می‌شود:

• پاک‌سازی و استانداردسازی[1] متن از قبیل اصلاح نویسه[2]ها، فاصله و نیم‌فاصله‌ها، وند[3]ها یا کلمات مرکب[4] جدا نوشته‌شده، ریشه‌یابی[5] و ...

• غنی‌سازی یا حاشیه‌نویسی متن (افزودن اطلاعات جانبی مفید به متن)؛ از قبیل برچسب‌زنی نقش ادات سخن[6]، گسترش معنایی کلمات[7] (افزودن کلمات هم‌معنی[8] یا هم‌کاربرد با کلیدواژه[9]های داخل متن) و ...

• حذف ویژگی‌های اضافه (کلمات بدون‌ارزش) از قبیل حذف کلمات توقف (ایست‌واژه‌ها)[10]، انتخاب کلیدواژه‌ها یا موجودیت‌های نامی و حذف سایر کلمات.

در پژوهش حاضر، ابتدا متن یک‌دست (بهنجارسازی) شد و علائم نگارشی[11] حذف شدند. سپس، کلمات موجود در ابیات از هم تقطیع[12] شدند و ایست‌واژه‌ها نیز، بر اساس فایلی که شامل این کلمات بود، حذف گردیدند. این ایست‌واژه‌ها با توجه به داده‌های موجود انتخاب شده بودند.

اولین گام در دسته‌بندی متن این است که متون به صورتی مناسب که توسط الگوریتم یادگیری ماشین قابل‌استفاده باشد، تبدیل شوند. هر کلمه در متن یک ویژگی[13] محسوب

---

1. normalization
2. character
3. affixes
4. compound words
5. stemming
6. part-of-speech tagging
7. semantic expansion
8. synonym
9. keyword
10. stop words
11. punctuation
12. named entity
13. feature

می‌شود، اما از آنجا که با افزایش مجموعهٔ داده‌ها، تعداد بالای ویژگی‌ها مشکل‌آفرین خواهد شد، سعی بر آن است تا ابعاد داده‌ها تا حدممکن کاهش یابد.

هدف از کاهش ابعاد[1] چیست؟

- پیشگیری از مزاحمت ابعاد (زمانی که تعداد ویژگی‌ها زیاد می‌شود، خطای دسته‌بندی داده‌ها افزایش پیدا می‌کند)؛
- کاهش مدت زمان و حافظهٔ استفاده‌شده توسط الگوریتم‌های داده‌کاوی؛
- تسهیل بصری‌سازی[2] داده‌ها؛
- تسهیل حذف داده‌های بی‌ربط.

یکی از راه‌های کاهش ابعاد این است که به‌جای به‌کارگیری همه ویژگی‌ها، از زیرمجموعه‌ای از ویژگی‌های مهم استفاده کنیم. اگرچه ممکن است به نظر برسد که این روش باعث ازدست‌رفتن اطلاعات می‌شود، اما باید توجه داشت که در این روش، تنها ویژگی‌های غیرمرتبط و زائد را حذف می‌کنیم. بخشی از این هدف در پژوهش حاضر، با حذف ایست‌واژه‌ها محقق شد، اما از آنجا که تعداد داده‌های برچسب‌خورده اندک بودند، پس از حذف ایست‌واژه‌ها، از تمامی کلمات باقی‌مانده که ۲۱۰۷ مورد بودند، به‌عنوان ویژگی استفاده شد. همچنین، با توجه به صورت‌های خاص بعضی واژه‌ها در شعر که با شکل دستوری رایج متفاوت است، مشاهده شد که با ریشه‌یابی کلمات، دقت مدل تاحدی کاهش پیدا می‌کند. از این رو، در پژوهش حاضر از این کار صرف‌نظر شد.

از آنجا که اغلب روش‌های یادگیری ماشین از جمله دسته‌بندی متون، بر روی داده‌های عددی[3] قابل‌اجرا هستند، در گام دوم، داده‌های متنی باید به بردارهایی از اعداد تبدیل می‌شدند. به این فرایند، بردارسازی[4] یا بازنمایی برداری[5] متون گفته می‌شود. رویکردهای مختلفی برای بردارسازی متون در پردازش زبان طبیعی[6] وجود دارد:

---

1. dimension reduction
2. visualization
3. numerical
4. vectorization
5. vector representation
6. natural language processing

○ رویکرد سبد کلمات[۱] شامل:

• کدگذاری دودویی[۲] یا تک‌روشن[۳]؛

• بردارسازی شمارشی[۴] یا وزن‌دهی مبتنی بر «فراوانی اصطلاح - معکوس فراوانی سند»[۵]؛

• بردارسازی درهم[۶]؛

• شمارش چندگانه‌ها[۷]؛

○ رویکرد بازنمایی توزیع‌شده[۸] و انتقال یادگیری[۹]؛

• روش‌های تعبیهٔ کلمات و متن[۱۰] از قبیل Word2Vec، GloVe، Doc2Vec، FastText و ...؛

• روش‌های مبتنی بر یادگیری عمیق[۱۱] از قبیل لایهٔ تعبیه[۱۲] در شبکهٔ عصبی؛

---

۱. bag-of-words (BoW): مدلی ساده برای نمایش برداری متون که در آن ترتیب کلمات نادیده گرفته می‌شود و فقط به فراوانی هر کلمه توجه می‌شود.

۲. binary encoding: نوعی نمایش برداری که در آن فقط حضور یا عدم‌حضور یک ویژگی (مانند کلمه) در سند با مقدار ۰ یا ۱ مشخص می‌شود.

۳. one-hot encoding: شیوه‌ای برای نمایش برداری کلمات که در آن فقط یک عنصر مقدار ۱ دارد و بقیه صفر هستند.

۴. frequency/count vectorization: بردارسازی متن بر اساس شمارش فراوانی وقوع هر کلمه در یک سند.

۵. term frequency–inverse document frequency (TF-IDF): روشی برای وزن‌دهی به کلمات بر اساس فراوانی آنها در یک سند نسبت به فراوانی‌شان در کل مجموعهٔ اسناد، به‌منظور کاهش اهمیت کلمات پرکاربرد.

۶. hashing vectorization: بردارسازی با استفاده از توابع هش، بدون نیاز به ساخت واژگان کامل.

۷. gram: بردارسازی با استفاده از توالی‌های پیوسته‌ای از n کلمه یا نویسه با تأکید بر توالی و ترتیب قرارگیری آنها.

۸. distributed representation: روشی که در آن معنا و ویژگی‌های زبانی هر واحد (مانند کلمه) در میان چندین بعد از یک بردار پخش می‌شود، به‌گونه‌ای که شباهت‌های معنایی در فضای برداری حفظ می‌گردد.

۹. transfer learning: رویکردی در بردارسازی و یادگیری که در آن مدل‌های ازپیش‌آموزش‌دیده برای استخراج ویژگی‌های زبانی در وظایف جدید، بازاستفاده می‌شوند.

۱۰. word/document embeddig: نوعی نمایش متراکم و معنایی که روابط نحوی و معنایی را در فضای برداری حفظ می‌کند.

11. deep learning
12. embedding layer

• استفاده از بردارهای ازپیش‌آموزش‌دیده[1] از قبیل Elmo، GPT، BERT، GPT2، XLM، XLNet، Transform-XL، DistillBERT و ...؛

در این پژوهش، برای بردارسازی ابیات، یک بار از روش TF-IDF و بار دیگر از روش کدگذاری دودویی استفاده شد. به‌جز دو الگوریتم رگرسیون لجستیک و پرسپترون چندلایه که در دو روش فوق، حدود ده درصد در معیار دقت[2] اختلاف داشتند، سایر الگوریتم‌ها تفاوت قابل‌ملاحظه‌ای از خود نشان ندادند. درضمن، با توجه به تنوع کلمات از جهت ترتیب قرارگیری در اشعار و از سوی دیگر، تعداد کم داده‌ها، بردارسازیِ چندگانه در این پژوهش کارایی لازم را نداشت و از آن استفاده نشد.

در گام بعدی، باید داده‌ها به دو دستهٔ آموزش و آزمون تقسیم می‌شدند. با توجه به محدودیت تعداد داده‌های برچسب‌خورده، برای انتخاب بهترین الگوریتم که خطای کمتری داشته باشد، روش‌های بازنمونه‌گیری[3] مدنظر قرار گرفتند. یک راهکار موجود آن بود که زیرمجموعه‌ای از داده‌ها را به‌طور تصادفی انتخاب کنیم (مثلاً ۸۰ درصد از داده‌ها) و آموزش را روی آن انجام دهیم و از زیرمجموعهٔ کنارگذاشته‌شده (مثلاً ۲۰ درصد بقیه) برای آزمون الگوریتم پیاده‌سازی‌شده، استفاده کنیم. به تکرار این مراحل به نحوی که مدل چندین بار با ترکیب‌های مختلفی از داده‌های آموزشی و آزمون، آموزش داده شود، روش اعتبارسنجی متقابل[4] گفته می‌شود. هدف این روش، ارزیابی پایدارتر و دقیق‌تر از توانایی تعمیم مدل به داده‌های نادیده است و از بیش‌برازش[5] به داده‌های آموزش جلوگیری می‌کند. اگر در هر مرحله از تکرار آموزش، تنها یکی از داده‌ها برای آزمون کنار گذاشته شود و آموزش روی سایر داده‌ها انجام شود، به آن، اعتبارسنجی

---

۱. pretrained language model embedding: بردارسازی با استفاده از مدل زبانی ازپیش‌آموزش‌دیده که شامل ویژگی‌های معنایی و نحوی آموخته‌شده از حجم وسیعی از داده‌های متنی است.

2. accuracy

3. resampling

4. cross validation

۵. Overfitting: حالتی که مدل به‌خوبی داده‌های آموزشی را یاد می‌گیرد، اما در تعمیم به داده‌های جدید ناتوان می‌ماند.

متقابلِ تک‌نمونه‌ای[1] می‌گویند. در پژوهش حاضر، به‌منظور ارزیابی عملکرد مدل‌های یادگیری ماشین، از این روش اعتبارسنجی استفاده شده است.

دلیل انتخاب این روش، محدودبودن تعداد نمونه‌های آموزشی در مجموعه دادهٔ استفاده‌شده بود. در شرایطی که حجم داده‌ها کم است، تقسیم آن به بخش‌های جداگانه برای آموزش و آزمون می‌تواند منجر به کاهش قابل‌توجه دقت در ارزیابی شود، چرا که بخش بزرگی از داده‌ها صرف آزمون شده و مدل با اطلاعات ناکافی آموزش می‌بیند. LOOCV با کنار گذاشتن تنها یک نمونه در هر تکرار و استفاده از باقی‌ماندهٔ داده‌ها برای آموزش، این مشکل را تا حد زیادی برطرف می‌کند و امکان استفادهٔ حداکثری از داده‌های موجود را فراهم می‌سازد. همچنین، این روش ارزیابی دقیق‌تری نسبت به روش‌های ساده‌تر نظیر تقسیم ثابت داده‌ها ارائه می‌دهد، هرچند در مقایسه با آنها هزینهٔ محاسباتی بالاتری دارد.

در مرحلهٔ آموزش مدل‌ها، فرایند یادگیری مدل هربار برای یکی از الگوریتم‌های موردنظر، پیاده‌سازی شد. برای پیاده‌سازی و آموزش مدل، به جهت بهینه‌سازی کدنویسی، از کتابخانهٔ scikit-learn در زبان پایتون بهره گرفته شد. در این مرحله، از الگوریتم‌های نظارت‌شدهٔ[2] زیر استفاده شد:

• الگوریتم بیز ساده: این الگوریتم براساس قانون بیز و محاسبهٔ احتمالات شرطی و با فرض استقلال متغیرهای تصادفی بنا شده است. براساس نظریهٔ بیز، برای محاسبهٔ احتمال وقوع B به شرط A، الگوریتم ابتدا تمام مواردی را که رخدادهای A و B هم‌زمان اتفاق افتاده‌اند را حساب می‌کند و به تعداد رخدادهای تکی A، تقسیم می‌کند تا احتمال شرطی مدنظر محاسبه شود (محمودی سعیدآباد و سمیع‌زاده، ۱۳۹۷). در پژوهش حاضر، احتمال وقوع B، همان احتمال قرارگرفتن یک بیت در یکی از دو دستهٔ تلمیح‌دار یا بدون‌تلمیح است و A، همان بردار ویژگی‌ها (کلمات) است. این الگوریتم که یکی از موفق‌ترین

1. leave-one-out cross-validation (LOOCV)
2. supervised

الگوریتم‌های یادگیری ماشین محسوب می‌شود، در کارهایی مثل تشخیص هرزنامه‌ها[1] کاربرد بسیار دارد.

• الگوریتم ماشین بردار پشتیبان: این الگوریتم سعی می‌کند یک مرز مناسب پیدا کند که داده‌های متعلق به دسته‌های مختلف را با بیشترین فاصله از هم جدا کند. این مرز تصمیم‌گیری ممکن است یک خط (در فضای دوبعدی) یا صفحه (در فضای سه‌بُعدی) یا اَبَرصفحه[2] (در فضای با ابعاد بالاتر) باشد. داده‌هایی که نزدیک‌ترین فاصله را به این مرز دارند، بردارهای پشتیبان[3] نامیده می‌شوند و نقش کلیدی در تعیین موقعیت مرز دارند. مرز بین دو دسته، با توجه به شرایط زیر تعیین می‌شود:

• تمام نمونه‌های دستهٔ اول در یک طرف و تمام نمونه‌های دستهٔ دوم در سمت دیگر مرز واقع شوند.

• مرز تصمیم‌گیری به گونه‌ای باشد که فاصلهٔ نزدیک‌ترین نمونه‌های آموزشی هر دو دسته از یکدیگر، در راستای عمود بر مرز تصمیم‌گیری، در بیشترین مقدار ممکن خود باشد.

بدین‌ترتیب، دو ابرصفحهٔ موازی در دو طرف مرز تصمیم‌گیری ایجاد می‌شوند؛ به گونه‌ای که ابرصفحهٔ مرز، بیشترین فاصله را بین دو ابرصفحهٔ موازی ایجاد کند (محمودی سعیدآباد و سمیع‌زاده، ۱۳۹۷)

برای داده‌هایی که به‌صورت خطی قابل‌تفکیک نیستند، الگوریتم SVM با بهره‌گیری از توابع هسته‌ای[4] داده‌ها را به فضای ویژگی با ابعاد بالاتر نگاشت می‌کند؛ در این فضا، امکان جداکردن کلاس‌ها با یک مرز خطی فراهم می‌شود. این قابلیت موجب شده است که SVM در مواجهه با داده‌های پیچیده نیز، عملکرد موفقی داشته باشد. برای انعطاف‌پذیری بیشتر در دسته‌بندی، از متغیرهای کمکی استفاده می‌شود تا اجازه داده شود برخی نمونه‌ها درون ناحیهٔ حاشیه‌ای قرار گیرند، حتی اگر با جریمه روبه‌رو شوند.

---

1. spam detection
2. hyperplane
3. support vectors
4. kernel functions

در این راستا، پارامتر C نقش کلیدی دارد؛ این پارامتر میزان حساسیت مدل را نسبت به خطاهای طبقه‌بندی تنظیم می‌کند و به‌عنوان یک پارامتر تنظیم‌کننده[1] بین افزایش پهنای حاشیه و کاهش خطا مصالحه برقرار می‌سازد.

از آن جا که نگاشت صریح داده‌ها به فضای بُعد بالاتر می‌تواند از نظر محاسباتی هزینه‌بر باشد، توابع هسته‌ای به‌گونه‌ای طراحی شده‌اند که بدون نیاز به انجام نگاشت واقعی، محاسبهٔ ضرب داخلی بردارها در فضای جدید را ممکن می‌سازند. این ویژگی، افزون بر کاهش بار محاسباتی، امکان استفاده از فضاهای با ابعاد بسیار بالا یا حتی نامتناهی را نیز فراهم می‌کند و به‌لحاظ زمان و حافظه بسیار کارآمد است (بیرانوند برجله، ۱۳۹۰). تابع هسته می‌تواند یکی از انواع گوسی ، چندجمله‌ای یا سیگموئید باشد.

• الگوریتم درخت تصمیم: درخت تصمیم الگوریتمی است که با یک ریشه شروع شده و رشد می‌کند. هر گره[2] درخت، یک گره تصمیم است که براساس ویژگی انتخاب شده، داده‌ها را به شاخه[3]هایی تقسیم می‌کند. هر شاخه نیز ممکن است به یک گره تصمیم دیگر برسد و یا به یک برگ که مشخص‌کنندهٔ دستهٔ داده‌هاست، ختم شود (بیرانوند برجله، ۱۳۹۰). درخت تصمیم می‌تواند با الگوریتم‌های مختلف یادگیری ماشین مانند CART و ID3 و C4.5 طراحی شود. بیشتر الگوریتم‌های درخت تصمیم، حریصانه است؛ چرا که در هر گام در پی شکلی از جداسازی است که در ظاهر بهترین انتخاب ممکن است (مستقل از انتخاب‌های قبلی و بعدی).

برای یک مجموعهٔ آموزشی درخت‌های بسیاری می‌توان در نظر گرفت، اما مطلوب‌ترین درخت، دارای کم‌ترین تعداد گره و سادگی بیشتر در عملیات مقایسه است. ناخالصی[4] معیاری برای تشخیص میزان مطلوبیت درخت دسته‌بندی است. هر انشعاب در صورتی «خالص» است که هر بخش آن شامل نمونه‌های یک دسته باشد. فرایند آموزش درخت باید به گونه‌ای صورت گیرد که در هر گام میزان خلوص افزایش یابد. یکی از

1. regularization parameter
2. node
3. branch
4. impurity

معیارهای سنجش خلوص، آنتروپی اطلاعاتی[1] است. آنتروپی درواقع نشان‌دهندهٔ کم‌بودن اطلاعات است؛ یعنی در مجموعه دادهٔ موردنظر، چه میزان قادر به تشخیص کلاس نهایی هستیم.

• الگوریتم جنگل تصادفی: مبنای دسته‌بندی در الگوریتم جنگل تصادفی، ساخت مجموعه‌ای از درخت‌های تصمیم و رأی‌گیری میان آنها برای تعیین برچسب نهایی است؛ به‌طوری‌که دسته‌ای که بیشترین تعداد رأی را کسب کرده باشد، به‌عنوان خروجی انتخاب می‌شود. برای ایجاد تنوع در درخت‌های این مجموعه، معمولاً در هر بار رشد یک درخت، زیرمجموعه‌ای تصادفی از ویژگی‌ها انتخاب می‌شود تا روند آموزش کنترل شود. این الگوریتم بر دو ویژگی کلیدی استوار است: ویژگی نخست، استفاده از روش بگینگ[2] است. در این روش، برای هر درخت، مجموعه‌ای آموزشی با انتخاب تصادفی و با جایگزینی از مجموعهٔ آموزش اصلی تهیه می‌شود، سپس درخت تصمیم بر پایهٔ این مجموعهٔ جدید ساخته می‌شود. ویژگی دوم، انتخاب تصادفی ویژگی‌ها در هر گره از درخت است؛ به این صورت که در هر گره، تنها یک زیرمجموعهٔ تصادفی از کل ویژگی‌ها در نظر گرفته می‌شود و از میان آنها، ویژگی‌ای که بیشترین بهرهٔ اطلاعاتی[3] را دارد، برای تقسیم گره انتخاب می‌شود (امینی خویی، ۱۳۹۴).

• الگوریتم k ـ نزدیک‌ترین همسایه: در این روش فرض می‌شود که تمام نمونه‌ها نقاطی در فضای n بعدی حقیقی ویژگی‌ها هستند و همسایه‌ها بر مبنای فاصلهٔ اقلیدسی[4] تعیین می‌شوند. منظور از k تعداد همسایه‌های درنظرگرفته‌شده است. هنگامی که ورودی جدیدی ارائه می‌شود، دستهٔ نمونهٔ آموزشی‌ای که شبیه‌ترین نمونه بر اساس معیار فاصله به ورودی است، به‌عنوان دستهٔ ورودی در نظر گرفته می‌شود (امینی خویی، ۱۳۹۴). در پژوهش حاضر، با تعداد ۳ همسایه، نسبت به ۲ و ۱ همسایه، نتایج بهتری حاصل شد.

---

1. information entropy
2. bagging
3. information gain
4. Euclidean distance

• الگوریتم رگرسیون لجستیک: این روش یک مدل آماری برای پیش‌بینی مقدار متغیرهای وابستهٔ[1] دوسویه است. منظور از دوسویه‌بودن، رخداد یک واقعهٔ تصادفی در دو موقعیت ممکن است؛ مثل بیماری یا سلامت، زندگی یا مرگ، هرزبودن یک رایانامه یا عادی‌بودن آن، و یا در این پژوهش، تلمیح‌دار بودن یک بیت یا بدون‌تلمیح‌بودن آن. این مدل در انتخاب متغیرهای مستقل محدودیتی ندارد و می‌تواند با انواع داده‌ها به‌کار رود. هدف الگوریتم این است که با استفاده از تابع سیگموئید[2]، خروجی مدل به یک مقدار بین صفر و یک نگاشت شود. این الگوریتم به‌ویژه در طبقه‌بندی داده‌ها و تحلیل روابط بین متغیرهای مستقل[3] و وابسته کاربرد فراوان دارد.

پیچیدگی چنین مدل‌هایی، تابعی از تعداد و اندازهٔ پارامترهای آنهاست. افزایش بیش از حد این پیچیدگی ممکن است منجر به بیش‌برازش شود. برای جلوگیری از این مسأله، معمولاً یک عبارت جریمه[4] به تابع هزینه[5] افزوده می‌شود تا از بزرگ‌شدن بیش‌ازحد پارامترها جلوگیری کند. در رگرسیون لجستیک، تابع هزینه بر پایهٔ منفی لگاریتم درست‌نمایی[6] تعریف می‌شود تا با کمینه‌کردن آن، به بیشینه‌سازی تابع درست‌نمایی دست یابیم. این روش به‌دلیل سادگی، تفسیرپذیری و دقت مناسب، در حوزه‌های مختلفی از جمله علوم اجتماعی، زیستی و پزشکی کاربرد فراوان دارد.

• الگوریتم پرسپترون چندلایه: پرسپترون چندلایه یک مدل پیشخور[7] شبکهٔ عصبی مصنوعی است که مجموعه‌ای از داده‌های ورودی را به مجموعهٔ مناسب خروجی تصویر می‌کند. یک پرسپترون چندلایه شامل تعدادی لایه از گره‌ها در یک گراف جهت‌دار[8] است. این گراف بین یک لایه و لایهٔ بعدی کاملاً متصل است. به‌جز گره‌های ورودی، باقی گره‌ها نورون‌هایی (واحد پردازش در شبکهٔ عصبی) با تابع فعالیت غیرخطی[9] هستند.

---

1. dependent variable
2. sigmoid function
3. independent variable
4. regularization term
5. loss function
6. log-likelihood
7. feedforward
8. directed graph
9. nonlinear activation function

پرسپترون چندلایه از روش یادگیری نظارتی انتشار به عقب[1] برای آموزش شبکهٔ خود استفاده می‌کند. پرسپترون چندلایه حالت بهبودیافتهٔ پرسپترون خطی استاندارد است که قابلیت جداسازی داده‌هایی را دارد که به‌صورت خطی تفکیک‌پذیر نیستند (بیرانوند برجله، ۱۳۹۰).

در مرحلهٔ پایانی، برای ارزیابی عملکرد الگوریتم‌ها و مقایسهٔ آنها با هم، به‌جز معیار دقت که نسبت پیش‌بینی‌های درست به کل پیش‌بینی‌ها را نشان می‌دهد، از معیار اف ـ ۱[2] نیز استفاده شد. این معیار، در واقع یک نوع میانگین بین دو پارامتر ارزیابی صحت[3] و فراخوانی[4] است و معیار ترکیبی مفیدی است برای دسته‌های نامتوازن. در این پژوهش هم، چون تعداد داده‌های آموزشی در دو دستهٔ تلمیح‌دار و بدون‌تلمیح مساوی نیستند، این معیار عملکرد، ارزیابی بهتری را به ما نشان می‌دهد. همچنین در این پژوهش، از ماتریس درهم‌ریختگی[5] برای دست‌یافتن به تصویری جامع‌تر در ارزیابی عملکرد مدل استفاده گردید. شکل کلی این ماتریس در شکل ۲ آورده شده است.

شکل ۲: ماتریس درهم‌ریختگی

		دستهٔ پیش‌بینی‌شده	
		بدون‌تلمیح	**تلمیح‌دار**
دستهٔ واقعی	بدون‌تلمیح	TN	FP
	تلمیح‌دار	FN	TP

۱. backpropagation: الگوریتمی برای آموزش شبکه‌های عصبی است که با محاسبهٔ گرادیان خطا نسبت به وزن‌ها، آنها را به‌صورت تدریجی به‌روزرسانی می‌کند تا خطای خروجی کمینه شود.

2. f1-score
3. precision
4. recall
5. confusion matrix

٤. یافته‌ها

با توجه به نتایج مندرج در جدول ۲ و مقایسهٔ نتایج حاصل از ارزیابی عملکرد هفت الگوریتم موردمطالعه، ملاحظه شد که از نظر معیار دقت، دو الگوریتم بیز ساده و رگرسیون لجستیک نتیجهٔ بهتری از خود نشان می‌دهند. باوجوداین، این معیار، به‌تنهایی معیار دقیقی برای تصمیم‌گیری نیست؛ چراکه عواملی مانند نامتوازن‌بودن اندازهٔ داده‌های آموزشی در دو دسته، روی آن تأثیر می‌گذارد. به‌عنوان مثال، توجه به ماتریس درهم‌ریختگی نشان می‌دهد که الگوریتم جنگل تصادفی در تشخیص ابیات تلمیح‌دار بهتر عمل می‌کند، ولی در تشخیص ابیات بدون‌تلمیح آن‌قدرها موفق نیست و در مجموع عملکرد خوبی ندارد. در حالی که الگوریتم بیز ساده نسبت به بقیه، نتایج متعادل‌تری از خود نشان می‌دهد و اهمیت این موضوع در پژوهش حاضر، به این دلیل است که در پیکرهٔ آموزشی ما، تعداد داده‌های تلمیح‌دار از بدون‌تلمیح بیشتر است. از نظر زمان آموزش مدل با استفاده از اعتبارسنجی متقابل، مشاهده شد که الگوریتم بیز ساده از سرعت بسیار بالاتری نسبت به سایر الگوریتم‌ها برخوردار است و الگوریتم جنگل تصادفی از همهٔ روش‌های دیگر به زمان بیشتری برای آموزش مدل نیاز دارد.

جدول ۲: مقایسهٔ عملکرد الگوریتم‌ها

الگوریتم	زمان آموزش مدل با استفاده از اعتبارسنجی متقابل LOOCV (برحسب ثانیه)	دقت مدل (برحسب درصد) با استفاده از نمایش برداری TF-IDF	معیار اف ـ ۱ با استفاده از نمایش برداری TF-IDF	دقت مدل (برحسب درصد)، با استفاده از نمایش برداری one-hot	معیار اف ـ ۱ با استفاده از نمایش برداری one-hot	ماتریس درهم‌ریختگی
درخت تصمیم	٤٦٠/١١	٦٨/٠٤	٠/٧٧	٦٣/٢٦	٠/٧٣	١٠٣ ٥٧ ٢٣٤ ٦٦
جنگل تصادفی	٩٩٢/١٣	٦٨/٩١	٠/٨	٦٨/٢٦	٠/٨	١٣٦ ٢٤ ٢٩٠ ١٠
رگرسیون لجستیک	٥٩/٧٣	٦٦/٧٤	٠/٨	٧٦/٠٩	٠/٨٤	٩٣ ٦٧ ٢٨٣ ١٧
بیز ساده	١٢/٥٩	٧١/٥٢	٠/٨٢	٧٦/٠٩	٠/٨	٣٥ ١٢٥ ٢٢٥ ٧٥
ماشین بردار پشتیبان	٣٨٠/٣٣	٧٤/٧٨	٠/٨٣	٧٤/٣٥	٠/٨٢	٧٨ ٨٢ ٢٦٠ ٤٠
k ـ نزدیک‌ترین همسایه	١٥٥/٧٥	٧٢/٣٩	٠/٧٩	٦٩/٧٨	٠/٨	١٢٣ ٣٧ ٢٨٤ ١٦
پرسپترون چندلایه	٤٣٧/٥١	٦٥/٢٢	٠/٧٩	٧٥/٢٢	٠/٨٢	٧٤ ٨٦ ٢٦٠ ٤٠

جدول ۳: نمونه‌هایی از پیش‌بینی الگوریتم‌ها

پیش‌بینی	متن	در پس آینه طوطی‌صفتم داشته‌اند آنچه استاد ازل گفت بگو می‌گویم	شاه عالم را بقا و عز و ناز*باد و هر چیزی که باشد زین قبیل	ز شور و عربده شاهدان شیرین‌کار* شکر شکسته سمن ریخته رباب زده	چو منصور از مراد آنان که بردارند بر دارند بدین درگاه حافظ را چو می‌خوانند می‌رانند	بوی پیراهن گمگشتهٔ خود می‌شنوم* گر بگویم همه گویند ضلالی‌ست قدیم
	واقعیت	تلمیح‌دار	بدون‌تلمیح	بدون‌تلمیح	تلمیح‌دار	تلمیح‌دار
	جنگل تصادفی	تلمیح‌دار	بدون‌تلمیح	بدون‌تلمیح	تلمیح‌دار	تلمیح‌دار
	رگرسیون لجستیک	تلمیح‌دار	بدون‌تلمیح	بدون‌تلمیح	تلمیح‌دار	تلمیح‌دار
	بیز ساده	تلمیح‌دار	بدون‌تلمیح	بدون‌تلمیح	تلمیح‌دار	تلمیح‌دار
	ماشین بردار پشتیبان	بدون‌تلمیح	بدون‌تلمیح	بدون‌تلمیح	تلمیح‌دار	تلمیح‌دار
	k نزدیکترین همسایه	تلمیح‌دار	بدون‌تلمیح	بدون‌تلمیح	بدون‌تلمیح	بدون‌تلمیح
	درخت تصمیم	بدون‌تلمیح	بدون‌تلمیح	بدون‌تلمیح	بدون‌تلمیح	بدون‌تلمیح
	پرسپترون چندلایه	تلمیح‌دار	بدون‌تلمیح	بدون‌تلمیح	تلمیح‌دار	تلمیح‌دار

۵. بحث و نتیجه‌گیری

پژوهش حاضر با هدف بررسی عملکرد چند الگوریتم یادگیری ماشین در دسته‌بندی اشعار فارسی به دو گروهِ تلمیح‌دار و بدون‌تلمیح انجام شد. در این راستا، از الگوریتم‌های مختلف یادگیری نظارت‌شده، از جمله بیز ساده، ماشین بردار پشتیبان، درخت تصمیم، جنگل تصادفی، k نزدیک‌ترین همسایه، رگرسیون لجستیک و پرسپترون چندلایه استفاده شد و با بهره‌گیری از روش اعتبارسنجی متقابل LOOCV، عملکرد هر الگوریتم سنجیده شد.

در حالت کلی، مشکل اساسی در دسته‌بندی خودکار متون، حجم بالای ویژگی‌هایی است که از متن استخراج می‌شود که در بسیاری از الگوریتم‌ها موجب کندشدن عملکرد دسته‌بند و ناکارآمدی آن خواهد شد. از طرف دیگر، وجود کلماتی متفاوت که معنای مشابه دارند و یا کلمات یکسانی که معانی متفاوت دارند ، ثابت‌نبودن طول اسناد متنی و وابستگی موجود میان اجزای یک متن، مشکلات را دوچندان می‌کند. علاوه‌براین، ویژگی‌هایی وجود دارند که گاه نه‌تنها باعث دسته‌بندی بهتر اسناد نمی‌شوند، بلکه دقت دسته‌بندی را هم کم می‌کنند.

در زبان فارسی، توسعهٔ نظام دسته‌بندی خودکار متون فارسی به‌دلیل ماهیت زبان فارسی (صرف و نحو پیچیده و منعطف، وجود پسوندها و پیشوندهای متنوع، پوشیده‌بودن ساختار نحوی به‌دلیل حذف بعضی اجزای جمله در برخی موارد، گاه چندمعنایی واژه‌ها و ابهام معنایی) و دردسترس‌نبودن مجموعه‌ای دقیق شامل ریشهٔ کلمات[1] و از سوی دیگر، نبود مجموعه‌ای استاندارد برای آزمون، کاری نسبتاً دشوار به نظر می‌رسد.

غیر از موارد فوق، در مسألهٔ مربوط به پژوهش حاضر، مشکلات دیگری نیز به چشم می‌خورند، از جمله تنوع بسیار در واژه‌ها، چراکه هر شاعر بنابر ذوق و سبک مخصوص خود، واژه‌هایی جدید و گاه منحصربه‌خود، ابداع می‌کند و یا این که گاهی ترتیب معمول کلمات یک جمله را به هم می‌ریزد تا طرحی نو دراندازد. درنهایت، می‌توان گفت هر

---

۱. واژه‌ها در فارسی ممکن است ریشهٔ مشخص و ثابتی نداشته باشند یا ریشه‌یابی‌شان پیچیده باشد.

آنچه که اوج زیبایی هنری شعر را به نمایش می‌گذارد، برای یک الگوریتم دسته‌بندی ماشینی مایهٔ دردسر است!

در حوزهٔ تلمیح، یکی از مشکلات، اختلاف‌نظر در تعریف گسترهٔ تلمیح و تعیین مواردی است که در این حوزه قرار می‌گیرند. این مشکل میان دسته‌بندی انسانی و ماشینی مشترک است. از طرف دیگر، در برخی از اشعار، تلمیح با صراحت بیشتر و با اتکا به واژه‌هایی خاص صورت گرفته است که تشخیص آن را آسان‌تر کرده است و به نظر می‌رسد که کشف شباهت میان این اشعار برای ماشین، راحت‌تر از اشعاری است که تلمیح موجود در آنها شکلی ضمنی‌تر دارد و تشخیص آنها نیازمند تخصص‌های انسانی است. وجود انواع مختلف تلمیح (تلمیحات اسطوره‌ای، حماسی و ملّی، قرآنی و ...) هم باعث می‌شود که دسته‌بندی دشوارتر شود، چراکه دو بیت که یکی تلمیح قرآنی دارد و دیگری تلمیح اسطوره‌ای و ملّی، از نظر تنوع واژگانی فضاهایی کاملاً متفاوت دارند. موضوع مهم دیگر که از ذات زبان سرچشمه می‌گیرد، پویایی مصادیق تلمیح است که باعث می‌شود هر مدل دسته‌بندی در تشخیص تلمیح در اشعار جدیدتر با چالش مواجه شود. به نظر می‌رسد بهره‌گیری از مجموعه‌داده‌های گسترده‌تر و متنوع‌تر که انواع تلمیح را پوشش دهند، در کنار بهره‌گیری از روش‌های پیشرفته‌تر، مانند یادگیری انتقالی و مدل‌های زبانی ازپیش‌آموزش‌دیده، می‌تواند دقت این نوع دسته‌بندی را در آینده بهبود ببخشد.

تعارض منافع

تعارض منافع ندارم.

ORCID

Mohammad Bahrani https://orcid.org/0000-0003-1645-1562

Parisa Mohammadian Kalkhoran https://orcid.org/0000-0002-1468-797X

منابع

آذین، زهرا، بحرانی، محمد. (۱۳۹۳). شناسایی خودکار شاعران شعر نو با استفاده از ویژگی‌های سبکی. *مجموعه مقاله‌های همایش خاتم (دانشگاه علامه طباطبایی)*، به‌کوشش نعمت‌الله ایران‌زاده (۱ـ۱۶). تهران: دانشگاه علامه طباطبائی.
https://www.noormags.ir/view/fa/articlepage/1078911

امینی خویی، زهره. (۱۳۹۴). *طبقه‌بندی ترافیک شبکه با استفاده از الگوریتم‌های یادگیری ماشین*. پایان‌نامه کارشناسی ارشد، دانشگاه کردستان.

بیرانوند برجله، آزاده. (۱۳۹۰). *فیلتر کردن هرزنامه‌ها با استفاده از تکنیک‌های یادگیری ماشین*. پایان نامه کارشناسی ارشد، دانشگاه شهید چمران اهواز.

جوانمردی، کامیار، اکبری، منوچهر. (۱۳۹۷). روش‌های یادگیری ماشین در بررسی ویژگی‌های زبان شعری در اشعار شاعران دفاع مقدس (مطالعهٔ موردی: اشعار دو شاعر دفاع مقدس؛ قیصر امین‌پور و محمدرضا عبدالملکیان). *فصلنامهٔ علمی ـ ترویجی مطالعات دفاع مقدس*، ۴(۳)، ۱۲۱ـ۱۴۴. https://civilica.com/doc/1377460

داد، سیما. (۱۳۸۵). *فرهنگ اصطلاحات ادبی* (چاپ سوم). تهران: انتشارات مروارید.

رازی، شمس‌الدین محمدبن‌قیس. (۱۳۷۲). *المعجم فی معاییر اشعار العجم (جلد اول)*، به کوشش دکتر سیروس شمیسا. تهران: انتشارات فردوس.

شفیعی کدکنی، محمدرضا. (۱۳۸۷). *شعر معاصر عرب*. تهران: انتشارات سخن.

شمیسا، سیروس. (۱۳۸۱). *فرهنگ تلمیحات (اشارات اساطیری، تاریخی، مذهبی در ادبیات فارسی)*. تهران: انتشارات فردوس.

کزازی، میرجلال‌الدین. (۱۳۷۲). *رخسار صبح* (چاپ ششم). تهران: انتشارات مرکز.

مازندرانی، محمدهادی‌بن محمدصالح. (۱۳۷۵). *انوارالبلاغه*. به کوشش محمدعلی غلامی‌نژاد، تهران: انتشارات میراث مکتوب.

مجیری، محمدمهدی، مینایی بیدگلی، بهروز. (۱۳۸۷). تشخیص وزن عروضی اشعار فارسی، کاربرد جدیدی از متن‌کاوی. دومین کنفرانس داده‌کاوی ایران. تهران: ایران.
https://civilica.com/doc/70402

محبتی، مهدی. (۱۳۸۰). *بدیع نو، هنر ساخت و آرایش سخن*. تهران: انتشارات سخن.

محمودی سعیدآباد، الناز، سمیع‌زاده، رضا. (۱۳۹۷). کاربرد الگوریتم‌های یادگیری ماشین در متن‌کاوی با رویکرد آنالیز احساس. *مدیریت فناوری اطلاعات*،۱۰(۲)، ۳۰۹-۳۳۰. https://www.magiran.com/p1823367

یاحقی، محمدجعفر. (۱۳۸۸). *فرهنگ اساطیر و داستان‌واره‌ها در ادبیات فارسی* (چاپ دوم). تهران: انتشارات فرهنگ معاصر.

References

Azin, Z., & Bahrani, M. (2014). Automatic identification of modern Persian poets using stylistic features. In N. Iranzadeh (Ed.), *Proceedings of Allameh Tabataba'i University*, (pp. 1–16). Tehran: Allameh Tabataba'i University. [In Persian] https://www.noormags.ir/view/fa/articlepage/1078911

Amini Khoyi, Z. (2015). Network traffic classification using machine learning algorithms [Master's thesis, University of Kurdistan]. [In Persian]

Biranvand Barjleh, A. (2011). Spam filtering using machine learning techniques [Master's thesis, Shahid Chamran University of Ahvaz]. [In Persian]

Javanmardi, K., & Akbari, M. (2018). Machine learning methods in examining poetic language features in the poems of Sacred Defense poets (Case study: Ghaysar Aminpour and Mohammadreza Abdolmalekian). *Scientific-Promotional Quarterly of Sacred Defense Studies*, *4*(3), 121–144. [In Persian] https://civilica.com/doc/1377460

Dad, S. (2006). *Dictionary of Literary Terms* (3rd ed.). Tehran: Morvarid. [In Persian]

Kazazi, M. J. (1993). *The Face of Dawn* (6th ed.). Tehran: Markaz. [In Persian]

Mahmoudi Saeedabad, E., & Sami'Zadeh, R. (2018). Application of machine learning algorithms in text mining with a sentiment analysis approach. *Journal of Information Technology Management, 10*(2), 309–330. [In Persian] https://www.magiran.com/p1823367.

Mazandarani, M. H. M. S. (1996). *Anwar al-Balagha* (M. A. Gholaminajad, Ed.). Tehran: Mirath-e Maktub. [In Persian]

Mohabbati, M. (2001). *Modern Badi‘: The Art of Composition and Rhetorical Embellishment*. Tehran: Sokhan. [In Persian]

Mojiri, M. M., & Minaei Bidgoli, B. (2008). Prosodic meter recognition in Persian poetry: A new application of text mining. *Second Iranian Data Mining Conference*. Tehran: Iran. [In Persian] https://civilica.com/doc/70402

Razi, Sh. M. Q. (1993). *Al-Mu'jam fi Ma'ayir Ash'ar al-Ajam* (Vol. 1) (S. Shamisa Ed.). Tehran: Ferdows. [In Persian]

Shafi‘i Kadkani, M. R. (2008). *Modern Arabic Poetry*. Tehran: Sokhan. [In Persian]

Shamisa, S. (2002). *Dictionary of Allusions (Mythological, Historical, and Religious References in Persian Literature)*. Tehran: Ferdows. [In Persian]

Tizhoosh, H. R. (2008). Poetic features for poem recognition: A comparative study. *Journal of Pattern Recognition Research, 3*(1), 24-39. https://doi.org/10.13176/11.62.

Yahaghi, M. J. (2009). *Dictionary of Myths and Narratives in Persian Literature* (2nd ed.). Tehran: Farhang-e Mo’aser. [In Persian]

استناد به این مقاله: محمدیان کلخوران، پریسا و بحرانی، محمد. (۱۴۰۴). مقایسهٔ عملکرد الگوریتم‌های پایهٔ یادگیری ماشین در دسته‌بندی اشعار فارسی به دو گروه تلمیح‌دار و بدون‌تلمیح. *علم زبان*، ۱۲(۲۱)، ۴۵-۷۶. doi: 10.22054/ls.2021.60784.1453

Language Science is licensed under a Creative Commons Attribution-Noncommercial 4.0 International License.