https://jpll.ui.ac.ir/?lang=en
Research on Mystical Literature
E-ISSN: 2476-3292
Document Type: Research Paper
Vol. 19, Issue 1, No.54, 2025, pp. 145-169
Received: 27/05/2025 Accepted: 19/07/2025

Evaluating the Combination of Language Models and Classification Techniques for Improving the Classification of Persian Literary Prose

Reza Ramezani [-]
Associate Professor, Department of Software Engineering, Faculty of Computer Engineering, University of Isfahan, Isfahan, Iran
r.ramezani@eng.ui.ac.ir

Melika Khandan
B.A. Graduate, Department of Software Engineering, Faculty of Computer Engineering, University of Isfahan, Isfahan, Iran
Melika.khandan79@gmail.com

Samaneh Taheri
PhD Student of Persian Language and Literature, Department of Persian Language and Literature, Faculty of Literature and Humanities, University of Isfahan, Isfahan, Iran
samanehtaheri176@gmail.com

Abstract
The classification of classical Persian literary prose, characterized by complex linguistic structures and profound semantic layers, poses significant challenges in Natural Language Processing (NLP). This study evaluates the effectiveness of combining transformer-based language models and diverse classification techniques to enhance the thematic classification of Persian literary prose. Beyond employing conventional approaches, which include pre-trained models like mBERT, ParsBERT, and RoBERTa, the research introduces innovative strategies, such as integrating embeddings from multiple models and employing numerical token outputs for cross-model classification. Traditional embedding methods, including TF-IDF, Bag of Words, and FastText, are also utilized, with extracted vectors fed into classifiers like LSTM, GRU, SVM, Random Forest, and Logistic Regression. The novel integration of transformer embeddings with vector-based classifiers yields significant improvements in accuracy, recall, and F1-score. This approach not only advances text classification but also supports the development of intelligent systems for information retrieval and Persian literary analysis.

Keywords: Text Classification, Natural Language Processing, Persian Prose, Deep Learning, Transformer Models.

- Corresponding author
Ramezani, R. , Khandan, M. and Taheri, S. (2025). Evaluation of the Combination of Language Models and Classification Techniques for Improving the Classification of Persian Literary Prose. *Research on Mystical Literature, 19 (1): 145-169.*

10.22108/jpll.2025.145439.1926

Introduction

Classical Persian prose, especially mystical texts, contains a unique and rich blend of linguistic complexity and semantic depth, offering a window into Iranian cultural and philosophical traditions. However, limited studies have been conducted on the systematic analysis and classification of these texts within Natural Language Processing (NLP). Most existing NLP methods are designed for modern Persian and perform poorly when applied to classical texts with archaic vocabulary and intricate writing styles. The lack of annotated and balanced data further reduces the effectiveness of classifiers. Therefore, developing methods capable of handling these complexities is essential. This research proposes a novel and efficient approach to thematic classification of Persian literary prose, especially classical texts, by integrating deep learning architectures and various embedding techniques. The key innovation lies in the combination of different types of language models and classifiers to improve the system's performance and address the inherent challenges of classical Persian texts.

Review of Literature

In recent years, various approaches have been proposed for Persian text classification using either statistical models or deep learning architectures. Many studies have focused on contemporary applications such as sentiment analysis or topic labeling, without addressing the specific challenges of classical prose. For instance, Karimi and Shahrabadi (2019) utilized deep learning for sentiment classification in modern Persian, while Basiri and Kabiri (2017) introduced a sentence-level corpus to improve classification via Naive Bayes. Ahmadi et al. (2016) applied topic modeling methods like LDA and STC to outperform traditional Bag of Words, and Feizi-Derakhshi et al. (2022) demonstrated that BiLSTM with attention mechanisms enhances accuracy in Persian text classification. Furthermore, Farhoodi and Yari (2010) evaluated vector-based classifiers such as SVM and KNN using TF-IDF, emphasizing the role of feature representation.

Transformer-based models such as mBERT, ParsBERT, RoBERTa, and AraBERT have shown strong performance in Persian NLP, particularly in classification tasks. However, most of these studies have focused on modern texts and rarely examined the challenges posed by classical Persian prose, with its metaphorical depth, archaic language, and mystical themes.

This study builds on prior work by evaluating whether hybrid configurations—combining contextual embeddings with statistical features and diverse classifiers—can offer further improvements. Rather than replacing individual models, these combinations aim to complement their strengths and provide more robust performance for classifying semantically layered literary texts.

Methodology

The employed dataset consists of two merged subsets of old Persian prose texts, a total of 2,788 samples. After initial preprocessing (removal of duplicates and non-standard characters), multi-labeled samples were duplicated and restructured into single-label records, increasing sample count and enhancing class distribution. The Hazm library was used for the text normalization, tokenization, stopword removal, and lemmatization. For embedding, this study used TF-IDF, BoW, and FastText, as well as transformer-based models like mBERT, ParsBERT, AraBERT, and RoBERTa. Each embedding was combined with the following classifiers: transformer models, recurrent neural networks (LSTM and GRU), and classical machine learning models (SVM, Random Forest, and Logistic Regression). Various hybrid combinations were also constructed to evaluate performance. Evaluation was done using accuracy, precision, recall, F1-score (macro and weighted), and paired t-tests.

Results

The experimental results demonstrated that combined configurations generally provided broader improvements compared to their standalone transformer baselines. While the weighted F1-score did not always show substantial gains, many hybrid models consistently enhanced other metrics such as accuracy, precision, and recall. In particular, ParsBERT + FastText, mBERT + FastText, and ParsBERT + GRU each achieved stronger overall performance relative to their transformer baselines, with notable gains in accuracy, although their weighted F1-scores remained largely unchanged. The most consistent improvements were observed with RoBERTa-based combinations, which enhanced both weighted and macro F1-scores—for example, RoBERTa with mBERT embeddings raised the weighted F1 from 0.77 to 0.84 and the macro F1 from 0.56 to 0.72, while integration with ParsBERT embeddings similarly increased the weighted F1 to 0.84 and the macro F1 to 0.71.

The significance of these improvements was further supported by paired t-tests, confirming that a majority of the hybrid configurations achieved statistically meaningful gains over their corresponding standalone models. Among them, combinations such as ParsBERT + FastText ($t = -2.75$, $p = 0.0335$), AraBERT + GRU ($t = 4.52$, $p = 0.0040$), ParsBERT + LSTM ($t = 7.49$, $p = 0.0003$), mBERT + FastText ($t = -3.17$, $p = 0.0193$), and mBERT +

ParsBERT ($t = 12.85$, $p < 0.0001$) demonstrated particularly strong improvements, highlighting the consistency of hybrid models in enhancing classification performance, especially in thematically complex and imbalanced datasets.

Overall, the findings indicate that hybrid configurations are not only statistically significant but also practically effective in the thematic classification of classical Persian prose. Their ability to capture subtle semantic and stylistic features makes them particularly valuable for underrepresented classes, where conventional models often fall short. This enhanced representational capacity is especially important for texts characterized by intricate vocabulary, layered themes, and rhetorical complexity, as it enables more accurate and semantically grounded categorization. Beyond the Persian context, the proposed framework also offers potential for application to other linguistically rich and structurally complex languages, providing a robust foundation for more precise and context-aware literary analysis.

نشریهٔ علمی پژوهش‌های ادب عرفانی
سال نوزدهم، شمارهٔ اول، پیاپی ٥٤، ١٤٠٤، صص. ١٤٥-١٦٩
تاریخ وصول: ١٤٠٤/٣/٦ تاریخ پذیرش: ١٤٠٤/٤/٢٨

مقاله پژوهشی

ارزیابی ترکیب مدل‌های زبانی و روش‌های دسته‌بندی برای بهبود طبقه‌بندی نثرهای ادبی کلاسیک فارسی[1]

رضا رمضانی[-]، دانشیار دانشکده مهندسی کامپیوتر، دانشگاه اصفهان، اصفهان، ایران
r.ramezani@eng.ui.ac.ir
ملیکا خندان، فارغ‌التحصیل کارشناسی دانشکده مهندسی کامپیوتر، دانشگاه اصفهان، اصفهان، ایران
Melika.khandan79@gmail.com
سمانه طاهری، دانشجوی دکتری زبان و ادبیات فارسی، دانشکده ادبیات، دانشگاه اصفهان، اصفهان، ایران
samanehtaheri176@gmail.com

چکیده

طبقه‌بندی متون فارسی، به‌ویژه نثرهای ادبی کلاسیک که سرشار از ساختارهای زبانی پیچیده و لایه‌های معنایی عمیق هستند، یکی از چالش‌های کلیدی در پردازش زبان طبیعی[2] محسوب می‌شود. این پژوهش با هدف ارزیابی روش‌های مختلف یادگیری ماشین و مدل‌های زبانی گوناگون برای طبقه‌بندی موضوعی نثرهای ادبی فارسی انجام شده است. در این مطالعه افزون‌بر شیوه‌های رایج طبقه‌بندی (مانند استفاده از مدل‌های از پیش‌آموزش‌دیده مانند mBERT، ParsBERT و RoBERTa)، از رویکردهای نوآورانه‌ای نیز بهره گرفته شده است. این رویکردهای نوین شامل ترکیب تعبیه‌سازی[3]های دو مدل با هم یا استفاده از توکن‌های عددی استخراج‌شده از یک مدل برای طبقه‌بندی با مدل دیگر هستند که با هدف بهینه‌سازی عملکرد و تجزیه‌وتحلیل نتایج به کار رفته‌اند. همچنین، تکنیک‌های سنتی تعبیه‌سازی همچون TF-IDF، Bag of Words و FastText به کار گرفته شدند و برای ارزیابی عملکرد، بردارهای استخراج‌شده به مدل‌های متنوع طبقه‌بندی، ازجمله مدل‌های شبکه عصبی بازگشتی LSTM و GRU و همچنین مدل‌های طبقه‌بندی برداری (مانند SVM[4]، Random Forest و Logistic Regression) ارائه شدند. نوآوری این پژوهش در ترکیب تعبیه‌های مدل‌های ترنسفورمری با یکدیگر و با بردارهای ویژگی دیگر یا طبقه‌بندی آن‌ها با مدل‌های طبقه‌بندی‌برداری است که نتایج به‌دست‌آمده، بهبود معنادار معیارهای صحت، بازخوانی و امتیاز F1 را نشان می‌دهد. این رویکرد، افزون‌بر بهبود طبقه‌بندی متون فارسی، امکان شناسایی الگوهای زبانی و معنایی در نثرهای کلاسیک فارسی را فراهم می‌کند. همچنین، این پژوهش می‌تواند نقش مدل‌های زبانی و الگوریتم‌های یادگیری ماشین را در مطالعات ادبی گسترش دهد و آن‌ها را با نیازهای خاص متون ادبی سازگارتر و ازنظر عملکرد، دقیق‌تر از روش‌های پیشین سازد.

واژه‌های کلیدی: طبقه‌بندی متن، پردازش زبان طبیعی، نثر فارسی، یادگیری عمیق، مدل‌های ترنسفورمری.

---

۱. مقالهٔ حاضر برگرفته از پایان‌نامهٔ کارشناسی ملیکا خندان با عنوان «طبقه‌بندی موضوعی خودکار متون ادبی» در گروه مهندسی نرم‌افزار، دانشگاه اصفهان، دانشکدهٔ مهندسی کامپیوتر است.

- مسؤول مکاتبات

رمضانی، رضا، خندان، ملیکا و طاهری، سمانه . (١٤٠٤). ارزیابی ترکیب مدل‌های زبانی و روش‌های دسته‌بندی برای بهبود طبقه‌بندی نثرهای ادبی کلاسیک فارسی، *پژوهش‌های ادب عرفانی*، ١٩(١):١٤٥ -١٦٩.

10.22108/jpll.2025.145439.1926

2. NLP
3. Embedding
4. Support Vector Machine

۱. مقدمه

نثرهای ادبی کلاسیک فارسی، به‌ویژه متون عرفانی، گنجینه‌ای بی‌مانند از فرهنگ، تاریخ و حکمت ایرانی را در خود جای داده‌اند. این متون با ساختارهای زبانی پیچیده، الگوهای نحوی پیچیده و لایه‌های معنایی غنی، نه‌تنها منبعی ارزشمند برای پژوهشگران، دانشجویان و علاقه‌مندان به ادبیات فارسی به شمار می‌روند، بلکه دریچه‌ای به مفاهیم عمیق عرفانی مانند عشق، محبت و وحدت وجود می‌گشایند. بااین‌حال، تلاش‌ها و تحقیقات اندکی برای تحلیل و طبقه‌بندی سیستماتیک این متون در حوزهٔ پردازش زبان طبیعی انجام شده است. فقدان ابزارها و روش‌های پیشرفته برای تحلیل متون ادبی فارسی، دسترسی و استفاده از این منابع ارزشمند را برای جامعهٔ علمی محدود کرده است. باتوجه‌به اینکه بسیاری از روش‌های پردازش زبان طبیعی موجود برای متون عمومی و مدرن فارسی طراحی شده‌اند، این روش‌ها در تحلیل نثرهای ادبی کلاسیک که دارای واژگان کهن و سبک‌های پیچیده نگارشی هستند، عملکرد مطلوبی ندارند. افزون‌بر این، کمبود داده‌های برچسب‌گذاری‌شده و نامتوازن‌بودن مجموعه‌های داده، دقت مدل‌های طبقه‌بندی را کاهش می‌دهد. بنابراین، توسعه روش‌هایی که بتوانند این پیچیدگی‌ها را مدیریت کرده و دقت مدل‌های طبقه‌بندی را بهبود دهند، از اهمیت بالایی برخوردار است.

هدف اصلی این پژوهش، ارائهٔ روشی نوآورانه و کارآمد برای طبقه‌بندی موضوعی و مفهومی متون ادبی فارسی است. این مطالعه به‌طور ویژه بر نثر قدیمی فارسی تمرکز دارد و از معماری‌های یادگیری عمیق و طیف متنوعی از تکنیک‌های تعبیه‌سازی بهره می‌برد. این پژوهش با ترکیب مدل‌های زبانی و تکنیک‌های دسته‌بندی، به‌دنبال رفع چالش‌های پیچیدهٔ متون فارسی است. نوآوری کلیدی این تحقیق در رویکرد ترکیبی آن نهفته است؛ رویکردی که مدل‌های مختلف با کاربردهای متفاوت را با یکدیگر ترکیب می‌کند. برخلاف روش‌های متداول که فقط از یک مدل برای استخراج معنا استفاده می‌کنند، این مطالعه از ترکیب مدل‌های مختلف بهره می‌برد. این پژوهش نه‌تنها دقت تحلیل نثرهای کلاسیک فارسی را افزایش می‌دهد، بلکه با به‌کارگیری مدل‌های زبانی پیشرفته و روش‌های یادگیری ماشین، چشم‌اندازی تازه برای فهم بهتر روابط معنایی، مفاهیم زبانی و عرفانی ادبیات پارسی می‌گشاید. این رویکرد، امکان بهره‌گیری از رویکردهای میان‌رشته‌ای را در این حوزه فراهم می‌سازد.

در ادامهٔ این مقاله، ابتدا چهارچوب نظری و پژوهش‌های مرتبط بررسی می‌شود. سپس، روش‌شناسی تحقیق و مدل‌های پیشنهادی به تفصیل شرح داده خواهند شد. درنهایت، نتایج حاصل از آزمایش‌ها ارائه و تحلیل می‌شوند.

۲. چهارچوب نظری

۲ـ۱. مقدمه

این پژوهش با ترکیب مدل‌های مبتنی‌بر یادگیری عمیق و روش‌های سنتی پردازش متن، تلاش می‌کند دقت طبقه‌بندی متون ادبی فارسی را افزایش دهد. درحالی‌که پژوهش‌های پیشین عمدتاً از مدل‌های ترنسفورمری یا روش‌های آماری به‌صورت مجزا استفاده کرده‌اند، این جستار بر چگونگی رفع ضعف‌های روش‌های منفرد ازطریق ترکیب این رویکردها تمرکز دارد. در ادامهٔ این بخش، مدل‌های یادگیری عمیق و روش‌های سنتی پردازش متن که مبنای این تحقیق هستند، معرفی خواهند شد و مبانی نظری ترکیب این روش‌ها نیز بررسی می‌شود.

۲ـ۲. مدل‌های ترنسفورمری

مدل‌های ترنسفورمری نوعی معماری یادگیری عمیق هستند که به‌جای پردازش ترتیبی داده‌ها، امکان درک هم‌زمان تمام توکن‌های یک توالی را فراهم می‌کنند و به همین دلیل در تحلیل روابط معنایی و ساختاری میان کلمات بسیار کارآمد هستند (Mo et al, 2024). ویژگی اصلی این مدل‌ها، استفاده از مکانیسم توجه چندسری[1] است که امکان تمرکز بر بخش‌های مختلف متن را به‌صورت موازی می‌دهد. معماری ترنسفورمری در ابتدا برای ترجمهٔ ماشینی طراحی شد، اما به‌سرعت در طیف گسترده‌ای از وظایف پردازش زبان طبیعی ازجمله طبقه‌بندی متن، شناسایی نهادهای نامدار، خلاصه‌سازی و تولید متن به کار گرفته شد (De Vries et al., 2019). مدل‌های شناخته‌شده‌ای مانند BERT، GPT و RoBERTa براساس این معماری توسعه یافتند و به‌دلیل صحت و کارایی بالا، به یک استاندارد در حوزهٔ پردازش زبان طبیعی تبدیل شدند. این پژوهش نیز از مدل‌های ترنسفورمری به‌منظور بهره‌گیری از توانایی بالای آن‌ها در درک متون پیچیده، به‌ویژه متون ادبی فارسی، استفاده کرد.

- **مدل ParsBERT**

ParsBERT یک مدل تک‌زبانه است که بر پایهٔ معماری BERT و به‌طور خاص برای زبان فارسی توسعه یافته است. این مدل با استفاده از بیش از دو میلیون سند آموزش‌دیده و برای انجام وظایفی مانند طبقه‌بندی نثرهای ادبی فارسی کاربرد دارد (Farahani et al, 2021).

- **مدل AI-BERT**

AI-BERT نیز مدل تک‌زبانه‌ای است که به‌طور خاص برای زبان عربی طراحی شده است. این مدل روی مجموعه داده‌های بزرگ عربی آموزش دید و برای وظایفی همچون دسته‌بندی متن، ترجمهٔ ماشینی و پاسخگویی به پرسش‌ها استفاده می‌شود (Antoun et al., 2020). در این پروژه، از AI-BERT برای طبقه‌بندی نثرهای ادبی فارسی استفاده شد؛ زیرا متون قدیمی فارسی در مجموعهٔ داده شامل جملات و واژگان عربی هستند. افزون‌بر این، این مدل به‌منظور بررسی عملکرد آن در پردازش متون فارسی قدیمی و مقایسه با دیگر مدل‌ها در طبقه‌بندی نثرهای ادبی فارسی ارزیابی می‌شود.

- **مدل mBERT**

mBERT مدل برت چندزبانه است که برای ۱۰۴ زبان، ازجمله زبان فارسی، آموزش داده شده است. این مدل برای درک معنای کلمات در متون مختلف طراحی شده و به‌دلیل وجود کلمات عربی در متون قدیمی فارسی، برای طبقه‌بندی نثرهای ادبی فارسی نیز مناسب تشخیص داده شده است (Devlin et al, 2019).

- **مدل RoBERTa**

RoBERTa نسخه‌ای بهبودیافته از مدل BERT است که توسط تیم Facebook AI معرفی شد. این مدل با اعمال تغییراتی در فرایند آموزش و بهینه‌سازی‌های مختلف، مانند حذف وظیفهٔ پیش‌بینی جملهٔ بعدی، عملکرد بهتری در وظایف پردازش زبان طبیعی از خود نشان می‌دهد. RoBERTa برای طبقه‌بندی متن و شناسایی نهادهای نامدار کارآمدتر از BERT است (Liu et al., 2019).

این مدل با استفاده از داده‌های گسترده‌تر و تکنیک‌های بهینه‌شده آموزش دیده است که می‌تواند به بهبود دقت در طبقه‌بندی نثرهای ادبی فارسی کمک کند.

---

[1]. Multi-Head Attention

۲ـ۳. مدل‌های شبکه عصبی بازگشتی

شبکه‌های عصبی بازگشتی[1] نوعی از شبکه‌های عصبی مصنوعی هستند که برای پردازش داده‌های ترتیبی و وابسته به زمان طراحی شده‌اند (Rumelhart et al, 1986). برخلاف شبکه‌های عصبی معمولی که ورودی‌ها را به‌صورت مستقل پردازش می‌کنند، دارای حافظه‌ای داخلی هستند که اطلاعات قبلی را برای تحلیل بهتر توالی‌های داده حفظ می‌کنند. این ویژگی باعث می‌شود که شبکه‌های عصبی بازگشتی در وظایفی مانند ترجمهٔ ماشینی، تحلیل سری‌های زمانی، تشخیص گفتار و پردازش زبان طبیعی بسیار موثر باشند (Xue et al, 2018). این مدل‌ها قابلیت حفظ ارتباط بین کلمات در جملات طولانی را دارند؛ ازاین‌رو، برای پردازش متون ادبی فارسی که ویژگی‌هایی همچون افعال مرکب مجزا و ساختارهای نحوی پیچیده دارند و نیازمند پردازش وابستگی‌های طولانی‌مدت هستند، گزینه‌ای مناسب به حساب می‌آیند.

≠ مدل LSTM

LSTM[2] نوع خاصی از شبکه‌های عصبی بازگشتی است که برای حل مشکلات ناپایداری گرادیان در داده‌های ترتیبی توسعه یافته است. LSTM با استفاده از دروازه‌های ورودی، خروجی و فراموشی، وابستگی‌های بلندمدت در داده‌ها را یاد می‌گیرد. این مدل در کار با داده‌های ترتیبی مانند متن و صدا بسیار کارآمد است و برای وظایفی همچون ترجمهٔ ماشینی و تجزیه‌وتحلیل متن استفاده می‌شود (Hochreiter & Schmidhuber, 1997).

≠ مدل GRU

مدل GRU[3] مشابه LSTM است؛ اما با ساختاری ساده‌تر و پارامترهای کمتری طراحی شده است. این مدل از دو دروازه به‌روزرسانی و بازنشانی برای یادگیری وابستگی‌های بلندمدت استفاده می‌کند و به‌دلیل مصرف کمتر از حافظه، مدل بهینه‌تری نسبت به LSTM است. GRU نیز در پردازش داده‌های ترتیبی مانند متن و صدا مؤثر است و در وظایفی مانند ترجمهٔ ماشینی و تجزیه‌وتحلیل متن کاربرد دارد (Cho et al., 2014).

ازآنجایی‌که مدل‌های ترنسفورمری در پردازش روابط معنایی توالی‌های کوتاه موفق هستند، اما ممکن است در درک ارتباطات بلندمدت دچار چالش شوند؛ ازاین‌رو، استفاده از شبکه‌های عصبی بازگشتی مانند LSTM و GRU به‌عنوان رویکرد مکمل استفاده شد.

۲ـ۴. تکنیک‌های مبتنی‌بر بردارهای ویژگی

تکنیک‌ها و مدل‌های مبتنی‌بر بردارهای ویژگی یکی از روش‌های اصلی برای نمایش متون به‌صورت عددی در پردازش زبان طبیعی محسوب می‌شوند. این مدل‌ها داده‌های متنی را به بردارهای عددی تبدیل می‌کنند که اطلاعات آماری یا معنایی متن را در قالبی قابل‌استفاده برای الگوریتم‌های یادگیری ماشین ذخیره می‌کنند. هدف اصلی این روش‌ها، ساده‌سازی نمایش متن به شکلی است که ماشین‌ها بتوانند آن را به‌راحتی پردازش و تحلیل کنند.

این روش‌ها به‌طورکلی به دو دستهٔ اصلی تقسیم می‌شوند:

---

1. RNN
2. Long Short-Term Memory
3. Gated Recurrent Unit

۱. تکنیک‌های ساده آماری: این دسته شامل تکنیک‌هایی مانند Bag of Words[1] و TF-IDF است که براساس تکرار کلمات در متن عمل می‌کنند. در تکنیک Bag of Words ترتیب کلمات نادیده گرفته می‌شود و تعداد تکرار هر کلمه در یک متن محاسبه و به‌عنوان ویژگی‌های آن متن استفاده می‌شود. بدین‌ترتیب، هر متن به‌صورت یک ماتریس از بردارهای کلمات نمایش داده می‌شود که تعداد تکرار هر کلمه در آن متن را نشان می‌دهد (Manning et al , 2008).

اما تکنیک TF-IDF، یک تکنیک وزنی برای اندازه‌گیری اهمیت کلمات در متن است که از دو معیار «تعداد تکرار کلمه در متن» (TF) و «معکوس تعداد تکرار کلمه در مجموعه متون» (IDF) استفاده می‌کند. ترکیب این دو مقدار، با مشخص‌کردن اهمیت نسبی کلمات در متن، یک ماتریس از بردارهای کلمات ایجاد می‌کند. این مدل برای ارزیابی اهمیت کلمات در متون مختلف استفاده می‌شود.

تکنیک‌های ساده آماری، سریع و تفسیرپذیر هستند؛ اما روابط معنایی و نحوی را به‌طور کامل در نظر نمی‌گیرند.

۲. مدل‌های مبتنی‌بر یادگیری توزیعی: دسته‌ای شامل مدل‌های تعبیه‌سازی مانند FastText است که از روش‌های یادگیری عمیق برای ایجاد بردارهایی غنی‌تر و دقیق‌تر بهره می‌برند. این مدل‌ها قادر به درک روابط معنایی و نحوی بین کلمات هستند (Joulin et al, 2016) و معمولاً در تحلیل متون پیچیده عملکرد بهتری دارند. افزون‌بر این، FastText به‌ویژه در شبیه‌سازی کلمات نادر و ترکیبی، عملکرد بهتری نسبت به مدل‌های دیگر دارد؛ زیرا قادر است بردارهای کلمات را براساس نواحی متنی که کلمات در آن‌ها ظاهر می‌شوند، ایجاد کند. این ویژگی باعث می‌شود که این مدل‌ها در وظایف پردازش زبان طبیعی کارآمدتر عمل کنند.

۲ـ۵. مدل‌های یادگیری ماشین نظارت‌شده

مدل‌های یادگیری ماشین نظارت‌شده به الگوریتم‌هایی اطلاق می‌شود که برای انجام وظایف پیش‌بینی از داده‌های برچسب‌دار استفاده می‌کنند. در این مدل‌ها، داده‌های آموزشی شامل نمونه‌هایی هستند که ویژگی‌ها و برچسب‌های مربوطه (نتایج یا دسته‌ها) به آن‌ها اختصاص داده شده است. هدف اصلی این الگوریتم‌ها، یادگیری روابط و الگوهای پیچیده موجود در داده‌ها به‌منظور پیش‌بینی یا دسته‌بندی داده‌های جدید است. مدل‌های یادگیری ماشین نظارت‌شده به دو دستهٔ کلی تقسیم می‌شوند: مدل‌های دسته‌بندی (که هدف آن‌ها تخصیص داده‌ها به گروه‌های مختلف است) و مدل‌های رگرسیون (که هدف آن‌ها پیش‌بینی مقادیر عددی است). در این پژوهش، از مدل‌های Random Forest، SVM و Logistic Regression برای مقایسهٔ عملکرد روش‌های یادگیری ماشین سنتی با مدل‌های یادگیری عمیق استفاده شده است.

مدل Random Forest ≠

مدل Random Forest یک الگوریتم یادگیری ماشین است که هم برای دسته‌بندی و هم برای رگرسیون استفاده می‌شود (Breiman, 2001). این مدل با ترکیب چندین درخت تصمیم‌گیری که به‌صورت تصادفی و با استفاده از داده‌های آموزشی ساخته شده‌اند، عمل می‌کند. درنهایت، نتایج درخت‌ها با یکدیگر ترکیب و پیش‌بینی نهایی ارائه می‌شود. این مدل به‌دلیل استفاده از بگینگ و مقاوم بودن در برابر بیش‌برازش، صحت بالا و کاهش واریانس دارد. همچنین، می‌تواند اهمیت ویژگی‌ها را ارزیابی کند.

---

[1]. BoW

≠ **مدل SVM**

مدل SVM یکی دیگر از الگوریتم‌های قدرتمند یادگیری نظارت‌شده به شمار می‌رود که در حوزه‌های دسته‌بندی و رگرسیون کاربرد فراوانی دارد. این الگوریتم با تعیین یک ابرصفحهٔ بهینه، تلاش می‌کند تا داده‌ها را به‌گونه‌ای از یکدیگر تفکیک کند که فاصلهٔ مرزی بین کلاس‌ها حداکثر شود (Cortes & Vapnik, 1995). در شرایطی که داده‌ها به‌صورت خطی قابل جداسازی نباشند، SVM با بهره‌گیری از توابع کرنل، داده‌ها را به فضایی با ابعاد بالاتر نگاشت می‌کند تا امکان تفکیک‌پذیری آن‌ها فراهم شود. SVM صحت بالایی در دسته‌بندی داده‌های با ابعاد بالا دارد و در برابر بیش‌برازش مقاوم است.

≠ **مدل Logistic Regression**

مدل Logistic Regression یک الگوریتم یادگیری ماشین برای دسته‌بندی دودویی است (Hosmer Jr et al, 2013). این مدل از تابع سیگموید برای پیش‌بینی احتمال تعلق داده‌ها به یک دستهٔ خاص استفاده می‌کند. آموزش این مدل با استفاده از روش‌های بهینه‌سازی مانند کاهش گرادیان انجام می‌شود. رگرسیون لجستیک ساده، کارآمد و تفسیرپذیر است و معمولاً برای مسائل دسته‌بندی دودویی کاربرد دارد.

۶ـ۲. معیارهای ارزیابی

معیارهای ارزیابی به‌منظور سنجش دقت و کارایی مدل‌ها در مسائل طبقه‌بندی و مشابه آن به کار می‌روند. انتخاب معیار مناسب به ویژگی‌های خاص داده‌ها و اهداف مدل بستگی دارد. این معیارها به پژوهشگران امکان می‌دهند تا عملکرد مدل را پیش از اعمال آن بر داده‌های جدید، ارزیابی و بهینه‌سازی کنند.

≠ **صحت**[1]**:** ساده‌ترین معیار ارزیابی است. این معیار، نسبت تعداد پیش‌بینی‌های صحیح به تعداد کل پیش‌بینی‌های انجام‌شده برای یک مجموعه داده را نشان می‌دهد. صحت به‌طورکلی برای ارزیابی عملکرد مدل‌ها استفاده می‌شود و یکی از معیارهای استفاده‌شده برای ارزیابی مدل‌ها در این پژوهش نیز صحت است. روش محاسبهٔ این معیار در رابطهٔ (۱) نشان داده شده است:

$$Accuracy = \frac{Number\ of\ correct\ predictions}{Total\ number\ of\ predictions}$$ رابطهٔ ۱

≠ **بازخوانی**[2]**:** این معیار نسبت مثبت‌های واقعی[3] شناسایی‌شده توسط مدل را به کل نمونه‌های مثبت واقعی موجود در داده‌ها نشان می‌دهد. به عبارت دیگر، بازخوانی نشان می‌دهد که مدل چقدر قادر است نمونه‌های مثبت را به‌درستی شناسایی کند. هرچقدر مدل تعداد پیش‌بینی‌های مثبت نادرست (مثبت کاذب[4]) بیشتری داشته باشد، بازخوانی آن کاهش می‌یابد. در این پژوهش، از این معیار برای ارزیابی توانایی مدل‌ها در شناسایی صحیح کلاس‌های نثرهای ادبی استفاده شده است. نحوه محاسبه این معیار در رابطهٔ (۲) ارائه شده است

---

[1]. Accuracy
[2]. Recall
[3]. True positive
[4]. False positive

$$Recall = \frac{TP}{TP+FN}$$ رابطهٔ ۲

≠ **دقت**[1]: این معیار نسبت مثبت‌های واقعی را به تمام مثبت‌های پیش‌بینی‌شده توسط مدل نشان می‌دهد. هرچه این مدل مثبت کاذب بیشتری را پیش‌بینی کند، دقت کمتری دارد. این معیار به‌ویژه برای مجموعه داده‌های نامتوازن مفید است؛ زیرا توانایی ارائهٔ ارزیابی دقیق‌تری از عملکرد مدل در شرایط توزیع نابرابر کلاس‌ها دارد. فرمول محاسبه این معیار در رابطهٔ (۳) آمده است:

$$Precision = \frac{TP}{TP+FP}$$ رابطهٔ ۳

≠ **F1-Score**: یک معیار واحد است که دقت و بازخوانی را ترکیب می‌کند (میانگین وزنی). هرچه امتیاز F1 بالاتر باشد، عملکرد مدل بهتر است (Chicco & Jurman, 2020) . یک مدل، تنها در صورتی امتیاز F1 بالایی دریافت می‌کند که هم دقت و هم بازخوانی آن مقادیر بالایی داشته باشند. نحوهٔ محاسبهٔ این معیار در رابطهٔ (۴) آمده است:

$$F1 = \frac{2 \times precision \times recall}{precision+recall}$$ رابطهٔ ۴

۱) **F1–ماکرو (Macro-F1)**: میانگین امتیاز F1 را برای تمامی کلاس‌ها بدون توجه به توزیع نمونه‌ها در هر کلاس محاسبه می‌کند. در این روش، تمامی کلاس‌ها به یک اندازه در محاسبهٔ امتیاز تأثیر دارند. این معیار به‌ویژه در مجموعه‌داده‌های نامتوازن که توزیع نمونه‌ها میان کلاس‌ها نابرابر است، برای ارزیابی عملکرد مدل اهمیت دارد؛ زیرا از تمرکز بیش از حد روی کلاس‌های پُرنمونه جلوگیری می‌کند.

۲) **F1–وزنی (Weighted-F1)**: این معیار مشابه به Macro-F1 است، با این تفاوت که در آن برای محاسبه میانگین، وزن هر کلاس براساس تعداد نمونه‌های آن در مجموعهٔ داده در نظر گرفته می‌شود. این روش در مجموعه داده‌های نامتوازن که تعداد نمونه‌ها در کلاس‌های مختلف تفاوت زیادی دارد، عملکرد بهتری از خود نشان می‌دهد؛ زیرا کلاس‌های با تعداد نمونه بیشتر وزن بیشتری در محاسبه امتیاز دارند (Opitz & Burst, 2019).

باتوجه‌به نامتوازن بودن مجموعه‌داده، از هر دو معیار Macro-F1 و Weighted-F1 در کنار دقت، صحت و بازخوانی در نمودارهای ارزیابی مدل‌ها استفاده شد، تا ارزیابی دقیقی از عملکرد مدل‌ها در دسته‌بندی‌های مختلف ارائه شود.

≠ **آزمون T زوجی (Paired T-test)** : آزمون t زوجی، یک روش آماری پارامتریک است که برای مقایسهٔ میانگین دو گروه مستقل به کار می‌رود. این آزمون ارزیابی می‌کند که آیا تفاوت میان میانگین‌ها ازنظر آماری معنادار است یا صرفاً ناشی از نوسانات تصادفی در داده‌ها است. این آزمون در حوزهٔ طبقه‌بندی متن، ابزاری مؤثر برای تحلیل تفاوت عملکرد مدل‌ها براساس معیارهایی نظیر دقت، بازخوانی و امتیاز F1 به شمار می‌رود (Cacciarelli et al., 2024).

در این پژوهش، به‌منظور بررسی معناداری آماری تفاوت عملکرد میان مدل‌های منفرد و مدل‌های ترکیبی، از آزمون t استفاده شده است.

---

[1]. Precision

۳. پیشینه پژوهش

۳ـ۱. پژوهش‌های انجام‌شده

پژوهش کریمی و شهرآبادی (Karimi & Shahrabadi, 2019) نقش مهمی در پیشبرد پردازش زبان طبیعی فارسی، به‌ویژه با تمرکز بر استفاده از مدل‌های یادگیری عمیق در تحلیل احساسات، ایفا کرده است. آن‌ها باوجود محدودیت‌های موجود در منابع این حوزه، از مدل چندزبانه BERT (mBERT) برای طبقه‌بندی نقدهای فارسی به احساسات مثبت و منفی بهره بردند و در این مسیر با چالش‌هایی مانند ساختارهای زبانی پیچیده و محدودیت مجموعه‌داده‌ها مواجه بودند. تحقیق آن‌ها با مسئلهٔ عدم تعادل داده‌ها روبرو بود؛ به‌طوری‌که داده‌های موجود به‌سمت احساسات مثبت متمایل بودند. این عدم تعادل بر عملکرد mBERT تأثیر گذاشت و صحت ۰٫۴۹ و امتیاز F1 برابر با ۰٫۶۳ برای شناسایی احساسات منفی به دست آمد. این شاخص‌ها هم نقاط قوت مدل BERT در درک مفاهیم وابسته به متن در زبان فارسی را نشان می‌دهند و هم محدودیت‌های آن را برجسته می‌کنند: بدون وجود مجموعه‌داده‌های بزرگ‌تر و خاص فارسی، عملکرد این مدل محدود می‌ماند. درحالی‌که معماری mBERT در فهم جزئیات متنی برتر است، نبود داده‌های فارسی باکیفیت و حاشیه‌نویسی مناسب مانع از بهینه‌سازی کامل آن می‌شود. این تحقیق هم به‌عنوان نمونه‌ای از پتانسیل BERT برای تحلیل احساسات فارسی و هم به‌عنوان فراخوانی برای افزایش منابع داده در این زمینه عمل می‌کند.

بصیری و کبیری (Basiri & Kabiri, 2017) با توسعهٔ مجموعه‌داده SPerSent که بر تحلیل احساسات در سطح جمله متمرکز است و نیز واژه‌نامه CNRC، کمک شایانی به حوزهٔ تحلیل احساسات فارسی کردند. پژوهش آن‌ها به یک شکاف مهم در پردازش زبان طبیعی فارسی، یعنی کمبود داده‌های تحلیل احساسات در سطح جمله پرداخت. برای حل این مشکل، آن‌ها مجموعه‌ای شامل ۱۵۰،۰۰۰ جملهٔ فارسی را گردآوری کردند که هرکدام با نشانگرهای احساسی دودویی (مثبت یا منفی) و رتبه‌بندی علامت‌گذاری شده بودند. این مجموعه‌داده، منبعی اساسی برای تحلیل احساسات به زبان فارسی فراهم کرد که صحت و آموزش مدل‌ها را بهبود بخشید. با استفاده از طبقه‌بندی Naive Bayes، آن‌ها به نتایج جالب‌توجهی در تشخیص قطبیت (با صحت ۹۵٪) و پیش‌بینی رتبه‌بندی احساسات (با صحت ۹۲٪) دست یافتند. پژوهش آن‌ها بر اهمیت ایجاد منابع زبانی سفارشی برای رفع نیازهای خاص پردازش زبان طبیعی فارسی تأکید دارد. این مطالعه نه‌تنها اثربخشی منابعی مانند SPerSent و CNRC را نشان می‌دهد، بلکه ضرورت مجموعه‌داده‌ها و واژه‌نامه‌های خاص زبان برای بهبود نتایج NLP برای متون فارسی را برجسته می‌کند.

هاوارد و رادر (Howard & Ruder, 2018) با معرفی ULMFiT (تنظیم دقیق مدل زبان جهانی برای طبقه‌بندی متن) تأثیر یادگیری انتقالی را در پردازش زبان طبیعی نشان دادند. این روش با کاهش نرخ خطا بین ۱۸ تا ۲۴ درصد در شش مجموعه‌داده طبقه‌بندی متن، اهمیت تنظیم دقیق مدل‌های پیش‌آموزش‌دیده را برای بهبود عملکرد، به‌ویژه در زبان‌های کم‌منبع، برجسته کرد.

احمدی و همکاران (Ahmadi et al, 2016) رویکرد جدیدی را برای طبقه‌بندی متن‌های فارسی با استفاده از مدل‌های موضوعی ارائه می‌دهند تا محدودیت‌های روش سنتی کیسهٔ کلمات (BOW) را برطرف کنند. آن‌ها با به‌کارگیری مدل‌هایی مانند LDA و STC، بهبودهای چشمگیری در صحت طبقه‌بندی (تا ۲۹ درصد بهتر از BOW) به دست آوردند. روش آن‌ها از انسجام معنایی بین کلمات بهره می‌برد، هزینه‌های محاسباتی را کاهش می‌دهد و صحت را برای متن‌های فارسی افزایش می‌دهد.

فرهودی و یاری (Farhoodi & Yari, 2010) الگوریتم‌های SVM و KNN را برای طبقه‌بندی متن‌های فارسی با استفاده از مجموعه داده همشهری ارزیابی می‌کنند. آن‌ها اهمیت انتخاب ویژگی و تکنیک‌های بازنمایی برداری مانند TF-IDF و TFCRF را برجسته می‌کنند. یافته‌های آن‌ها نشان می‌دهد، درحالی‌که هر دو الگوریتم مؤثر هستند، KNN عملکرد بهتری دارد. آن‌ها همچنین بر نقش پیش‌پردازش، ازجمله حذف کلمات توقف [1]و توکن‌سازی[2]، در بهبود نتایج طبقه‌بندی تأکید می‌کنند.

فیضی و همکاران (۱۴۰۱) پژوهشی را در زمینهٔ طبقه‌بندی متون فارسی بر پایهٔ شبکه‌های عصبی عمیق ارائه می‌کنند که دو مدل شبکهٔ عصبی پیچشی [3]و شبکهٔ عصبی با حافظهٔ بلندمدت ـ کوتاه‌مدت دوسویه سلسله‌مراتبی [4]همراه با لایه توجه را برای این هدف توسعه می‌دهد. این مطالعه نشان می‌دهد که ParsBiLSTM، به‌دلیل قابلیت پردازش بهتر وابستگی‌های طولانی‌مدت در متون فارسی، عملکرد بهتری نسبت به ParsCNN دارد. همچنین، نتایج این پژوهش تأکید می‌کند که ترکیب شبکه‌های عمیق با لایه‌های توجه [5]می‌تواند به بهبود دقت طبقه‌بندی متون فارسی کمک کند.

۳ـ۲. مسائل حل‌نشده و مسیرهای آینده پژوهش

باتوجه‌به محدودیت‌های موجود در پردازش زبان طبیعی فارسی، پژوهش‌های گذشته اغلب با چالش‌های مختلفی روبرو بوده‌اند. در این پژوهش، هدف اصلی بهبود عملکرد مدل‌های زبانی و تکنیک‌های دسته‌بندی در تحلیل نثرهای ادبی فارسی با تمرکز بر نثرهای کلاسیک و ویژگی‌های معنایی خاص آن‌ها است. باوجود پیشرفت‌های بسیار در استفاده از مدل‌های مبتنی‌بر ترنسفورمر مانند BERT و سایر تکنیک‌های یادگیری ماشین در زبان‌های دیگر، هنوز مشکلاتی همچون کمبود منابع داده‌ای مناسب و پیچیدگی‌های خاص زبان فارسی بر روی نتایج تأثیرگذار هستند. این چالش‌ها به‌ویژه در نثرهای ادبی کلاسیک که ساختارهای معنایی پیچیده و غنی دارند، بیشتر نمایان می‌شوند. بنابراین، این پژوهش به‌دنبال ارزیابی ترکیب مدل‌های پیشرفته زبان و تکنیک‌های دسته‌بندی به‌منظور رفع این مشکلات و بهبود صحت طبقه‌بندی در متون ادبی فارسی است.

٤. روش پژوهش

رویکرد این پژوهش در طبقه‌بندی متن فارسی از یک چارچوب ساختاریافته شامل آماده‌سازی داده‌ها، پیش‌پردازش، تولید تعبیه‌سازی‌ها، آموزش مدل و ارزیابی پیروی می‌کند.

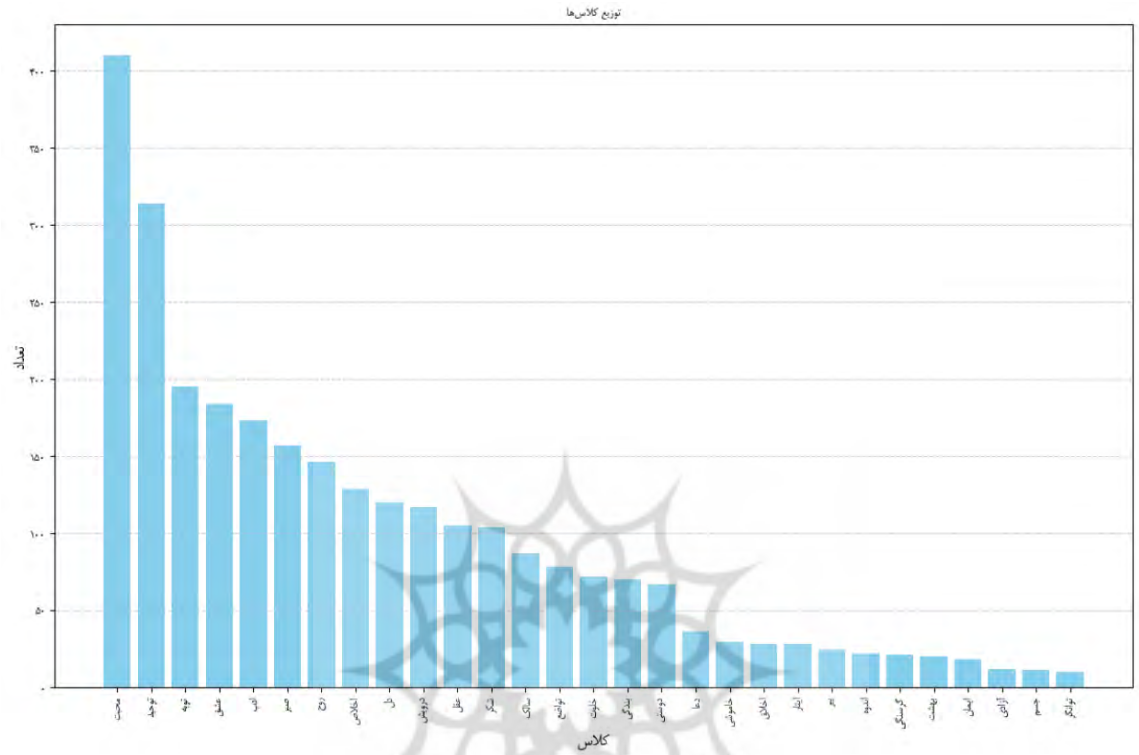
٤ـ۱.آماده‌سازی داده‌ها و پیش‌پردازش

داده‌های استفاده‌شده در این مطالعه شامل دو زیرمجموعه متشکل از ۷۰۰ و ۱۳۰۰ جمله یا متن کوتاه نثر فارسی قدیمی است. این متون کلمات عربی و برچسب‌های پراکنده و نامتوازن دارند. در مرحلهٔ اول، این دو زیرمجموعه با یکدیگر ترکیب و سپس تمام علائم غیرمتعارف و متون تکراری از داده‌ها حذف شدند. در ادامه، نثرهایی که دارای برچسب‌های

---

[1]. StopWord
[2]. Tokenization
[3]. ParsCNN
[4]. ParsBiLSTM
[5]. Attention Mechanism

دوتایی بودند (مانند برچسب عشق و محبت)، به دو نثر یکسان با دو برچسب متفاوت تبدیل شدند. همچنین، از میان برچسب‌های کلاس، آن‌هایی که تعداد نثرهایشان کمتر از ده عدد بود حذف شدند. این حذف به‌دلیل تأثیر منفی تعداد کم نمونه‌ها در یک کلاس بر فرایند تقسیم داده‌ها به مجموعه‌های آموزش، اعتبارسنجی و آزمون صورت گرفت.



شکل ۱. پراکندگی کلاس‌های مجموعه‌داده

در نهایت، یک مجموعه‌داده شامل ۲۷۸۸ نثر ادبی به دست آمد که برخی از نثرها میان دو یا چند کلاس مشترک هستند. برای مثال، نثر «چون ارادت قوی شد، نامش محبت گردد و چون محبت قوی شد، نامش عشق گردد، پس عشق نیست الّا محبت مفرط» در هر دو کلاس عشق و محبت وجود دارد.

بااین‌حال در این پژوهش، مسئله به‌صورت تک‌برچسبی[1] حل شد؛ به این معنا که نثرهایی که به بیش از یک کلاس تعلق داشتند، به چند نمونهٔ تکراری اما با برچسب‌های متفاوت تبدیل شدند. این رویکرد نه‌تنها تعداد کل نمونه‌ها را افزایش داد، بلکه توزیع داده‌ها را نیز متفاوت کرد. به‌ویژه، کلاس‌هایی با بیشترین هم‌پوشانی با سایر کلاس‌ها، پس از این فرایند شاهد افزایش حجم داده‌های خود بودند. این روش امکان استفاده از مدل‌های سنتی مانند SVM، Random Forest و Logistic Regression را بدون نیاز به تغییرات ساختاری پیچیده برای مسائل چندبرچسبی فراهم کرد.

شکل (۱) پراکندگی مجموعه‌داده و تعداد نمونه‌های هر کلاس را پس از این فرایند نشان می‌دهد. شکل (۲) نیز جدول داده[2] ایجادشده و بیشترین و کمترین مقدار کلمات در هر کلاس را نشان می‌دهد. بیشترین مقدار کلمه در میان ۲۷۸۷ نثر موجود در مجموعه‌داده، ۶۷ عدد و کمترین مقدار کلمه ۲ عدد است.

---

[1]. Single-label classification
[2]. Dataframe

	text	Class	Number of words	2	67
0	اخلاص بیرون کردن خلق است از معامله حق	اخلاص	8	NaN	NaN
1	...اخلاص سه قسم است اخلاص شهادت و آن در اسلام است	اخلاص	54	NaN	NaN
2	...الاخلاص فرض فی فرض و نقل فی نقل- گفت اخلاص ف	اخلاص	45	NaN	NaN
3	اخلاص اجابت است هر که را اجابت نیست، اخلاص نیست	اخلاص	10	NaN	NaN
4	...اخلاص آن است که بیرون آری خلق را از معامله خدا	اخلاص	18	NaN	NaN
...	...	...	...	...	...
2782	...سمنون، محبّت را بر معرفت تقدیم کردی و بیش ترین	محبت	36	NaN	NaN
2783	...هیچ مؤمن از اصل محبّت خالی نیست و لکن تفاوت از	محبت	54	NaN	NaN
2784	...قال بعض الحکماء من أعطی من المحبة شیئا فلم یعط	محبت	15	NaN	NaN
2785	حبّک للشیء یُعمی و یُصمّ	محبت	5	NaN	NaN
2786	...من تحقق حبه تحقق ایمانه و أنی یتحقق الإیمان بم	محبت	12	NaN	NaN

2787 rows × 5 columns

شکل ۲. جدول‌داده ایجادشده از مجموعه‌داده

پس از پالایش اولیه و ترکیب مجموعه‌داده‌ها، مرحلهٔ بعدی شامل پاک‌سازی و نرمال‌سازی متن‌ها است. این فرایند با هدف بهینه‌سازی داده‌ها برای پردازش، بدون ایجاد تغییر در ساختار واژگان (املا واژگان) یا سبک نگارشی متون ادبی کلاسیک برای حفظ اصالت متون انجام شده است. در این مرحله، از کتابخانهٔ هضم[1] استفاده شد که به‌دلیل تخصصی‌بودن در پردازش زبان فارسی، قابلیت‌هایی مانند یکدست‌سازی نویسه‌ها، استانداردسازی علائم و حذف نویسه‌های غیرضروری را فراهم می‌کند. این ابزار با افزایش دقت پردازش و کاهش ناخالصی داده‌ها[2]، عملکرد مدل‌های یادگیری ماشین را بهبود می‌بخشد.

۱. نرمال‌سازی متن

همهٔ فاصله‌های اضافی، کاراکترهای کنترلی یونیکد (مانند U+200C و U+200D)، نویسه‌های مخفی و هرگونه نشانهٔ نامرئی دیگر به‌طور کامل حذف شدند تا خوانایی متن بهبود یابد. برخی نویسه‌های غیراستاندارد (مانند «ۀ») به‌شکل استاندارد (مانند «ه») تبدیل شدند تا از پراکندگی داده‌ها جلوگیری شود.

۲. حذف علائم و نویسه‌های غیرضروری

همهٔ علائم نگارشی زائد (مانند «؛،:؟!»)، اعداد فارسی و انگلیسی و کاراکترهای لاتین که در پردازش متون ادبی کلاسیک کاربردی ندارند، حذف شدند. نشانه‌های عربی همچون «ًٌٍَُِّْ» که در متون قدیمی فارسی دیده می‌شوند اما تأثیری در پردازش ندارند، نیز حذف شده‌اند.

۳. حذف کلمات توقف[3]

از فهرست کلمات توقف کتابخانهٔ هضم برای حذف واژگان پرتکرار و کم‌ارزش استفاده شد. این فهرست به‌گونه‌ای تنظیم شده است که از حذف کلمات مهم جلوگیری کند.

---

[1]. Hazm
[2]. Data Noise
[3]. StopWord

۴. توکن‌سازی[1] و پردازش واژگان

متن‌ها به توکن‌های مستقل (کلمات) شکسته شدند تا امکان پردازش بهینه فراهم شود. فرایند ریشه‌یابی[2] و لماتیزه‌کردن[3] برای استخراج ریشه کلمات و بهبود عملکرد مدل انجام شد.

با اجرای این مراحل، مجموعه‌داده‌ای تمیز و استانداردسازی‌شده حاصل شد که در بخش بعدی برای استفاده در مدل‌های یادگیری ماشین آماده‌سازی خواهد شد. شکل (۳) تمامی مراحل انجام‌شده را نمایش می‌دهد.

شکل ۳. مراحل آماده‌سازی‌داده‌ها و پیش‌پردازش

٤ـ۲. تعبیه‌سازی و توکن‌سازی

در پردازش زبان طبیعی، استخراج ویژگی‌ها از متون یکی از گام‌های اصلی برای تحلیل داده‌های زبانی است. هر متن حاوی جملاتی است که از ترکیب کلمات به وجود آمده‌اند و کوچک‌ترین واحد پردازش محسوب می‌شوند. کلمات و بخش‌های مختلف متن به واحدهای کوچکتری به نام "توکن" تبدیل می‌شود. هر توکن ممکن است یک کلمه، بخشی از یک کلمه یا حتی نشانه‌گذاری‌های خاص باشد. هدف از توکن‌سازی این است که داده‌های غیرساختاری به اجزای قابل پردازش تبدیل شوند که می‌توانند به‌عنوان ورودی برای مرحلهٔ بعدی پردازش داده‌ها، یعنی تعبیه‌سازی استفاده شوند. در مرحلهٔ تعبیه‌سازی، این توکن‌ها به بردارهای عددی تبدیل می‌شوند. این بردارها نه‌تنها نمایانگر خود کلمه، بلکه نمایانگر ویژگی‌های معنایی و موضوعی آن کلمه نیز هستند. بنابراین، توکن‌ها پس از تبدیل به بردارهای تعبیه‌شده، به مدل‌های یادگیری ماشین ارائه می‌شوند؛ زیرا مدل‌های یادگیری ماشین قادر به پردازش داده‌های عددی هستند، این بردارهای تعبیه‌شده به مدل‌ها کمک می‌کنند تا روابط معنایی و الگوهای موجود در داده‌ها را شناسایی کند و درنتیجه، عملکرد مدل‌ها در وظایف مختلف مانند ترجمهٔ ماشینی، تحلیل احساسات و شناسایی موضوعات بهبود یابد.

رویکرد این پژوهش استفاده از چهار مدل مبتنی‌بر ترنسفورمر شامل mBERT، ParsBER، AraBERT و RoBERTa، به‌همراه سه تکنیک تعبیه‌سازی مبتنی‌بر بردار شامل Bag of Words (BOW)، TF-IDF و FastText است. ابتدا این روش‌های تعبیه‌سازی بر داده‌های پیش‌پردازش‌شده اعمال شدند تا نمایش‌های زبانی مختلفی تولید کنند. سپس، هریک از این تعبیه‌سازی‌ها در ترکیب با نُه طبقه‌بندی مختلف، ازجمله مدل‌های ترنسفورمر (mBERT ، ParsBERT، AraBERT ، RoBERTa)، معماری‌های عصبی بازگشتی (GRU، LSTM) و مدل‌های سنتی یادگیری ماشین (SVM ، Random Forest و Logistic Regression) استفاده شد. این ارزیابی جامع امکان تحلیل مقایسه‌ای برای شناسایی مؤثرترین ترکیب‌ها را فراهم کرد، که فراتر از روش‌های معمول مانند mBERT و ParsBERT است.

---

[1]. Tokenization
[2]. Stemming
[3]. Lemmatization.

٤ـ۲ـ۱. سازگاری میان مدل‌ها و تعبیه‌های ترکیبی ≠ دسته‌بندی با مدل‌های ترنسفورمری

در مدل‌های مبتنی‌بر ترنسفورمر مانند mBERT ، ParsBERT، AraBERT و RoBERTa، سازگاری میان مدل‌ها برای دسته‌بندی تعبیه‌های یک مدل با مدل دیگر، ازطریق تکنیک‌های ترکیب[1] تعبیه‌سازی انجام می‌شود. ازآنجاکه هر مدل ترنسفورمر از یک لغت نامه و واژگان منحصربه‌فرد استفاده می‌کند، واردکردن مستقیم توکن‌ها از یک مدل به مدل دیگر غیرممکن است و برای رفع این محدودیت از ترکیب تعبیه‌سازی‌ها استفاده می‌شود. به این صورت که تعبیه‌سازی‌های مدل‌های مختلف مانند (mBERT وParsBERT) به‌صورت موازی از هر دو مدل استخراج می‌شوند. سپس، با هم ترکیب می‌شوند و به‌عنوان ورودی به هرکدام از دو مدل mBERT وParsBERT به‌عنوان دسته‌بندی استفاده، داده می‌شوند.

افزون‌بر این، اگر دو مدل از معماری پایه‌ای یکسانی برخوردار باشند و لغت‌نامه‌هایشان تا حد زیادی مشابه باشد، می‌توان از نمایش‌های عددی توکن‌ها برای تبادل اطلاعات میان آن‌ها بهره برد. برای نمونه، برای دسته‌بندی با مدل mBERT که برای پردازش چندین زبان آموزش دیده است، می‌توان نمایش‌های عددی توکن‌های یک مدل دیگر مانند ParsBERT را استخراج و آن‌ها را در قالبی متناسب با نیازهای ورودی مدل mbert تنظیم کرد. این رویکرد به‌ویژه زمانی مؤثر است که هر دو مدل مبتنی‌بر معماری BERT باشند؛ زیرا در این صورت، احتمال شباهت توکن‌ها یا تجزیه آن‌ها به زیربخش‌های یکسان افزایش می‌یابد. این استراتژی‌ها امکان ادغام مؤثر چند مدل ترنسفورمر را فراهم می‌کنند و انعطاف‌پذیری و عملکرد سیستم طبقه‌بندی را بهبود می‌بخشند.

برای دسته‌بندی تعبیه‌سازی‌های مبتنی‌بر بردار با مدل‌های ترنسفورمری، بردارهای خروجی حاصل از روش‌های BOW و TF-IDF را نمی‌توان به‌طور مستقیم در مدل‌های ترنسفورمری به کار برد. این محدودیت‌ها به‌دلیل نمایش عددی ثابت، عدم حفظ ترتیب واژگان و تفاوت ماهوی در ساختار داده‌های ورودی است؛ زیرا بردارهای BOW و TF-IDF نمایشی کلی از کل متن در قالب یک بردار با طول ثابت ارائه می‌دهند که در آن ترتیب و معنای واژگان نادیده گرفته می‌شود. در مقابل، مدل‌های ترنسفورمری مانند BERT برای پردازش متن نیازمند توالی‌ای از توکن‌ها هستند که هر توکن با برداری با ابعاد ثابت نمایش داده می‌شود؛ اما برخلاف BOW و TF-IDF، طول این توالی بسته به میزان محتوای متن متغیر است. این تفاوت‌ها استفادهٔ مستقیم یا ترکیب این بردارها با بردارهای تعبیه‌شده مدل‌های ترنسفورمری را غیرممکن می‌سازد.

اما FastText، برخلاف TF-IDF و Bag of Words (BOW)، بردارهای معنایی پیوسته تولید می‌کند که امکان ترکیب با مدل‌های ترنسفورمری را فراهم می‌سازند. بااین‌حال، این ترکیب به‌صورت مستقیم امکان‌پذیر نیست. تفاوت اصلی FastText با مدل‌های ترنسفورمری مانند BERT در این است که FastText کلمات را در یک فضای برداری پیوسته نمایش می‌دهد؛ درحالی‌که مدل‌های ترنسفورمری از توکن‌های موقعیتی برای نمایش متن استفاده می‌کنند. افزون‌بر این، توکن‌های خاصی مانند [CLS] و [SEP] که در مدل‌های ترنسفورمری برای نگهداری اطلاعات معنایی و موقعیتی جملات به کار می‌روند، در FastText وجود ندارند. همچنین، تفاوت در ابعاد بردارهای خروجی این دو روش مانع از استفادهٔ مستقیم آن‌ها در مدل‌های ترنسفورمری می‌شود. برای حل این مشکل، ابتدا بردارهای FastText استخراج و در کنار توکن‌های مدل ترنسفورمری پردازش می‌شوند. سپس، باتوجه‌به تفاوت در طول بردارهای خروجی، همهٔ

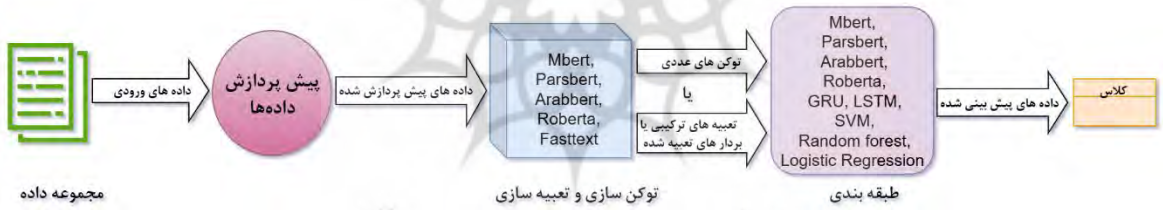
[1]. Concatenation

بردارها به یک طول ثابت تبدیل می‌شوند (با استفاده از پَد کردن یا حذف مقادیر اضافی). درنهایت، این بردارها با استفاده از لایهٔ اتصال[1] در شبکهٔ عصبی ترکیب و به‌عنوان ورودی به مدل ترنسفورمر ارائه می‌شوند. این ترکیب امکان بهره‌گیری هم‌زمان از ویژگی‌های معنایی FastText و قابلیت‌های توکنی ترنسفورمر را فراهم می‌کند و دقت دسته‌بندی متون را به‌طور چشمگیری بهبود می‌بخشد.

≠ دسته‌بندی با مدل‌های شبکه عصبی بازگشتی مانند GRU و LSTM

برای دسته‌بندی تعبیه‌سازی‌های مدل‌های ترنسفورمری (مانند BERT) با مدل‌های دنباله‌ای مانند GRU و LSTM، ابتدا از توکن‌سازی مدل مربوطه استفاده می‌شود. در این مرحله، توکن‌های خاص مانند [CLS] و [SEP] به متن اضافه شده و متن به طول ثابت (مثلاً ۵۰ توکن) برش یا تکمیل می‌شود. سپس، از آخرین لایهٔ مخفی مدل BERT (خروجی last_hidden_state) استفاده می‌شود که بردارهای تعبیه‌شده تولید می‌کند. این داده‌ها شامل input_ids (توکن‌ها) و attention_masks (ماسک توجه) هستند که به‌عنوان ورودی به مدل‌های شبکهٔ عصبی بازگشتی (GRU و LSTM)، برای طبقه‌بندی ارائه می‌شوند.

برای دسته‌بندی تعبیه‌سازی‌های مدل FastText با مدل‌های دنباله‌ای مانند GRU و LSTM، ابتدا توکن‌سازی با استفاده از Keras Tokenizer انجام می‌شود. در این مرحله، متن به توالی‌های عددی تبدیل شده و به طول ثابت برش یا تکمیل می‌شود. سپس، با استفاده از مدل FastText، بردارهای تعبیه‌شده با ابعاد ۳۰۰ برای هر کلمه استخراج می‌شوند. این بردارها به‌عنوان ورودی به مدل‌های GRU و LSTM، برای طبقه‌بندی ارائه می‌شوند.



شکل ٤. نمودار فرایند طبقه‌بندی با استفاده از تعبیه‌های ترکیبی مدل‌های ترنسفورمر و تکنیک‌های تعبیه‌سازی مختلف

≠ دسته‌بندی با مدل‌های یادگیری ماشین نظارت‌شده

مدل‌های یادگیری ماشین مانند SVM، Random Forest و Logistic Regression به ورودی‌های عددی نیاز دارند. این ورودی‌ها معمولاً در قالب بردارهای عددی (تعبیه‌ها) از متون استخراج می‌شوند. برای مدل‌های ترنسفورمری؛ هر متن با واژه‌ساز[2] مدل تعبیه‌ساز ورودی، به توکن (input_ids و attention_mask) تبدیل می‌شود سپس مدل، تعبیه‌های نهایی (از آخرین لایه یا میانگین تعبیه‌ها) را برمی‌گرداند. این بردارها به‌عنوان ویژگی (X) برای مدل استفاده می‌شوند. بردارهای ویژگی (Y) نیز با استفاده از تبدیل کلاس‌های متنی به مقادیر عددی با LabelEncoder تولید می‌شوند (درصورت نیاز از کاهش ابعاد بردارها و استانداردسازی داده‌ها با نرمال‌سازی ویژگی‌ها برای بهبود عملکرد مدل نیز استفاده می‌شود).

برای مدل FastText، پس از توکن‌سازی با استفاده از Keras، بردارهای تعبیه‌شده کلمات برای هر توکن، از مدل از

---

[1]. Concatenate
[2]. Tokenizer

پیش‌آموزش‌دیده FastText (مانند cc.fa.300.bin) استخراج می‌شوند. سپس، تعبیه‌های جملات با میانگین‌گیری روی بردارهای کلمات محاسبه‌شده و به‌عنوان ورودی به مدل‌های یادگیری ماشین ارائه می‌شوند.

برای تعبیه‌سازی‌های مبتنی‌بر TF-IDF و Bag of Words (BOW)، ابتدا متون ورودی به توکن‌های کلمه‌ای با استفاده از تابع word_tokenize تبدیل می‌شوند. در این فرایند، هر جمله به لیستی از کلمات تجزیه می‌شود. سپس برای کاهش تعداد کلمات منحصربه‌فرد ریشه‌یابی[1] و لماتیزه‌کردن[2] کلمات انجام می‌شود و در مرحله بعد، ویژگی‌های عددی متن با استفاده از دو روش رایج CountVectorizer (برای BOW) و TfidfVectorizer (برای TF-IDF) استخراج می‌شوند.

- CountVectorizer: این روش کلمات متن را براساس تعداد دفعات تکرار هر کلمه به بردارهای عددی تبدیل می‌کند.
- TfidfVectorizer: این روش افزون‌بر تعداد تکرار کلمات، به اهمیت هر کلمه در کل مجموعه‌داده نیز توجه می‌کند.

در نهایت، برای آماده‌سازی داده‌ها، کلاس‌های متون با استفاده از LabelEncoder از کتابخانه Scikit-Learn به مقادیر عددی تبدیل می‌شوند. سپس، این داده‌های عددی به مدل‌های شبکه عصبی بازگشتی و طبقه‌بندهایی مانند Support Vector Machines (SVM)، Random Forest و Logistic Regression ارائه می‌شوند تا فرایند دسته‌بندی انجام شود.

این تعبیه‌سازی‌های سنتی ابعاد کمتری دارند و به‌دلیل کارایی حافظه و زمان‌های آموزش سریع‌تر مؤثر بودند. طبقه‌بندهایی مانند Logistic Regression، SVM و Random Forest نیز با این تعبیه‌سازی‌های ساده و کم‌بُعد، عملکرد خوبی از خود نشان می‌دهند.

شکل ۵. نمودار فرایند طبقه‌بندی با استفاده از تعبیه‌های مدل‌های مبتنی‌بر بردارهای ویژگی

٤ـ۳. شرح یک نمونه ترکیبی موفق

در اینجا یکی از نمونه‌های ترکیبی موفق برای دسته‌بندی متون ادبی فارسی تشریح شده است:

مدل طبقه‌بندی بر پایهٔ BERT، با استفاده از TensorFlow پیاده‌سازی شده است. در این مدل، از یک رویکرد ترکیبی استفاده شده که تعبیه‌سازی‌های BERT و FastText را با یکدیگر ادغام می‌کند. این ترکیب برای درک عمیق تفاوت‌های معنایی و بافتی متن، به‌ویژه در زبان‌های غنی ازلحاظ مورفولوژیکی مانند فارسی، طراحی شده است. روند کار بدین‌صورت است که ابتدا متن ورودی توکن‌سازی می‌شود. سپس ID های توکن‌های BERT به لایهٔ TFBertModel منتقل می‌شوند تا تعبیه‌سازی‌های بافتی تولید شوند. این تعبیه‌سازی‌ها ویژگی‌های زبانی، روابط بین کلمات و ظرافت‌های معنایی را استخراج می‌کنند.

به‌صورت موازی مدل FastText نیز از فایل cc.fa.300.bin بارگذاری شده و تعبیه‌سازی‌های فارسی FastText استخراج می‌شوند. برای این منظور، تابعی ترکیبی (tokenize_and_embed) تعریف می‌شود که متن‌ها را توکن‌سازی و

[1]. stemming
[2]. lemmatization

بردارهای تعبیه‌شدهٔ FastText را برای آن‌ها استخراج می‌کند. برخلاف BERT ، FastText در سطح زیرکلمات عمل می‌کند و جزئیات مورفولوژیکی مانند پیشوندها، پسوندها و ریشهٔ کلمات را استخراج می‌کند. این ویژگی آن را به‌ویژه برای زبان فارسی که فرم‌های کلمات اغلب اطلاعات گرامری و معنایی قابل‌توجهی دارند، مناسب می‌سازد.

بردارهای خروجی تعبیه‌سازی‌های BERT و FastText ، همه به یک طول ثابت تبدیل می‌شوند (با پَد کردن یا حذف مقادیر اضافی) و با استفاده از یک لایه Concatenate در مدل TensorFlow ترکیب شده و به‌عنوان ورودی به لایه‌های طبقه‌بندی مدل BERT ارائه می‌شوند. این ترکیب، یک مجموعه ویژگی جامعی ایجاد می‌کند که مزایای هر دو نوع تعبیه‌سازی را در خود ادغام کرده است:

- BERT برای درک بافتی و نحوی.
- FastText برای جزئیات مورفولوژیکی و معنایی.

مدل بر روی مجموعه‌داده برچسب‌گذاری و آموزش داده شد و ۸۰٪ داده‌ها برای آموزش، ۱۰٪ برای اعتبارسنجی و ۱۰٪ برای تست استفاده شد. معیارهای ارزیابی کلیدی مانند صحت، دقت، بازخوانی و امتیاز F1 در طول آموزش و تست نهایی محاسبه شدند.

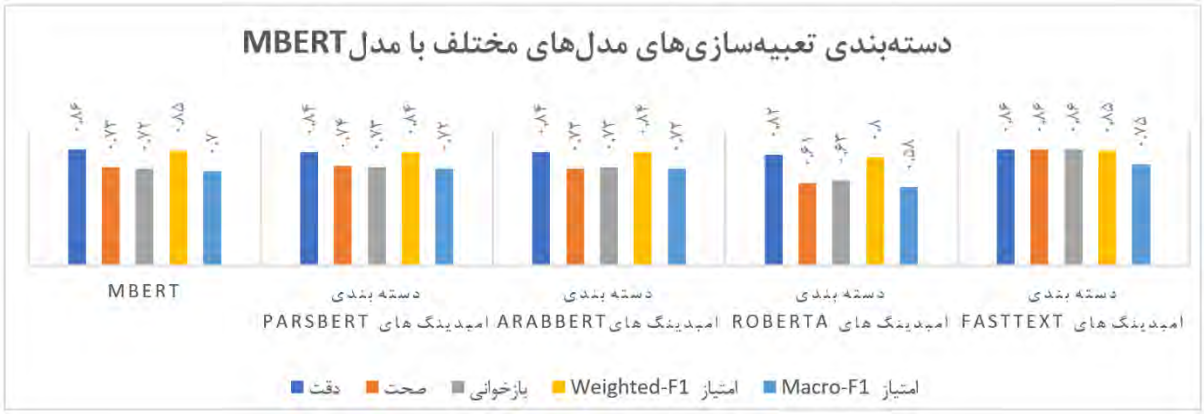
رویکرد تعبیه‌سازی ترکیبی، قدرت ترکیب ترنسفورمرهای حساس به بافت با مدل‌های برداری کارآمد برای طبقه‌بندی متن را نشان می‌دهد. این رویکرد یک استاندارد جدید را برای پردازش زبان‌های غنی ازلحاظ مورفولوژیکی مانند فارسی ایجاد می‌کند و راهکاری مقیاس‌پذیر، تطبیق‌پذیر و با عملکرد بالا ارائه می‌دهد. این مشارکت راه را برای کاربردهای پیشرفته در پردازش زبان طبیعی برای زبان‌های کم‌منبع هموار و تکنولوژی‌های زبان جامع‌تری را پدید می‌آورد.

۵. نتایج و تحلیل داده‌ها

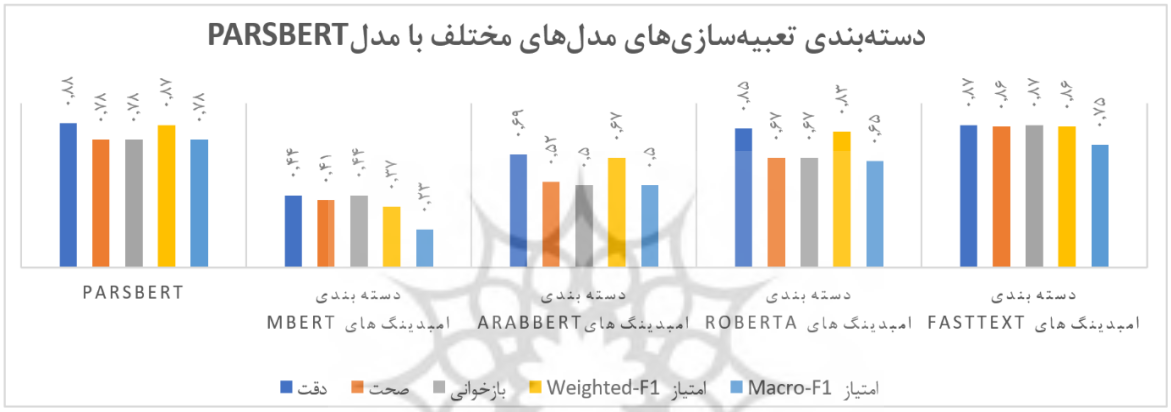
یکی از ویژگی‌های بارز مجموعه‌دادهٔ استفاده‌شده در این پژوهش، توزیع نامتوازن داده‌ها میان کلاس‌هاست؛ به‌طوری که برخی کلاس‌ها دارای نمونه‌های بسیار بیشتری نسبت به دیگر کلاس‌ها هستند. برای ارزیابی دقیق عملکرد مدل‌ها در چنین شرایطی، افزون‌بر معیارهای متداول مانند دقت، صحت و بازخوانی، از امتیاز-F1، در دو حالت معیارهای Macro-F1 و Weighted-F1 نیز استفاده شد. نتایج ارزیابی‌ها با امتیاز Macro-F1 نشان می‌دهد که باوجود توزیع نامتوازن داده‌ها، مدل‌ها در شناسایی هر دو گروه کلاس‌های پرتکرار و کم‌نمونه، عملکرد پذیرفته‌شده‌ای از خود نشان دادند. این نتیجه‌گیری بدون استفاده از روش‌های متداول متعادل‌سازی داده‌ها حاصل شد. میانگین اختلاف بین مقادیر Weighted-F1 و Macro-F1 معادل ۰,۱۲ بود که در محدودهٔ ۰,۰۵ تا ۰,۲ قرار می‌گیرد. در ادامه، عملکرد هر مدل در دسته‌بندی، تعبیه‌سازی مدل‌های مختلف در شکل‌های ۶ تا ۱۴ نمایش داده شده است.

شکل(۶) عملکرد مدل mbert را در دسته‌بندی تعبیه‌سازی‌های مدل‌های مختلف نمایش می‌دهد. نتایج حاکی از آن است که FastText در تمامی معیارها بهترین عملکرد را در مقایسه با مدل mBERT به‌تنهایی دارد. این موضوع، نشان‌دهندهٔ قدرت بالای تعبیه‌سازی‌های پیش‌پردازش‌شده در ترکیب با mBERT است.

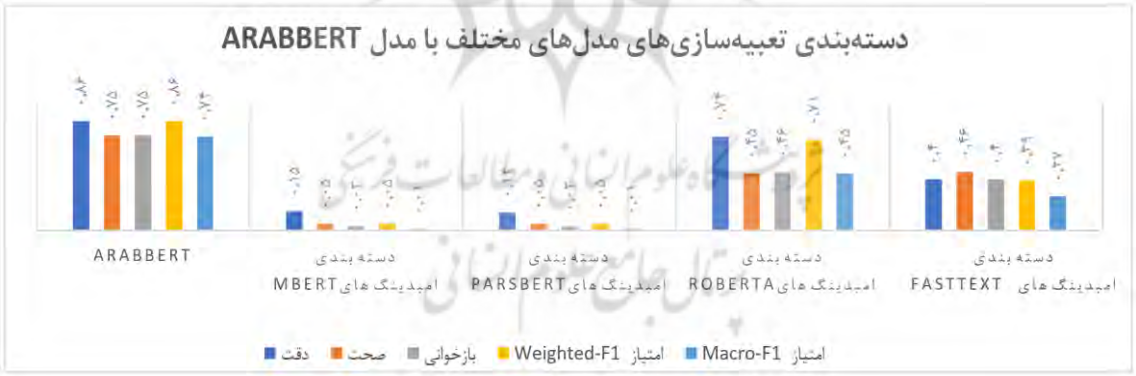
شکل (۷) نیز عملکرد مدل ParsBERT را در دسته‌بندی تعبیه‌سازی‌های مدل‌های مختلف نمایش می‌دهد. نتایج نشان می‌دهد که مدل ParsBERT در ترکیب با تعبیه‌های FastText، بهترین عملکرد را در مقایسه با مدل ParsBERT به‌تنهایی دارد.



شکل ٦. نمودار دسته‌بندی تعبیه‌سازی‌های مدل‌های مختلف با مدل mbert



شکل ۷. نمودار دسته‌بندی تعبیه‌سازی‌های مدل‌های مختلف با مدل ParsBERT



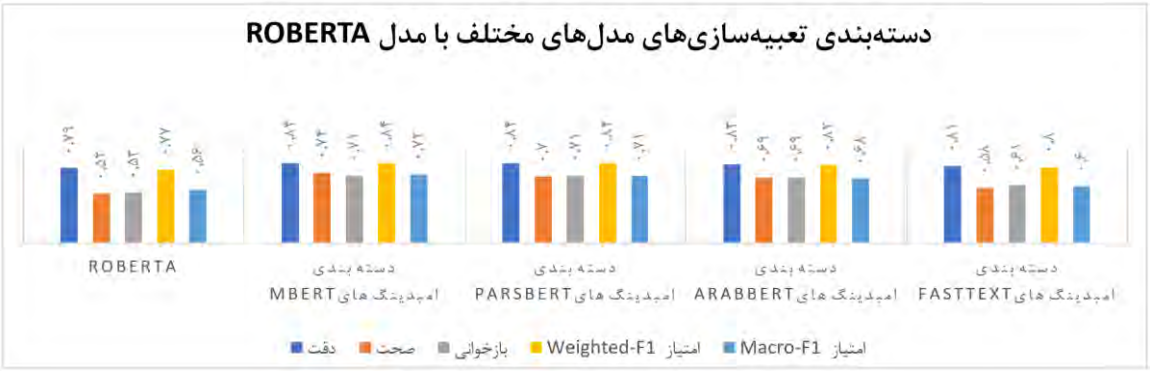
شکل ۸. نمودار دسته‌بندی تعبیه‌سازی‌های مدل‌های مختلف با مدل ArabBERT

شکل (۸) عملکرد مدل ArabBERT را در دسته‌بندی تعبیه‌سازی‌های مدل‌های مختلف نمایش می‌دهد. نتایج نشان می‌دهد که مدل عرب برت به‌تنهایی عملکرد بهتری نسبت به مدل‌های ترکیبی دارد.

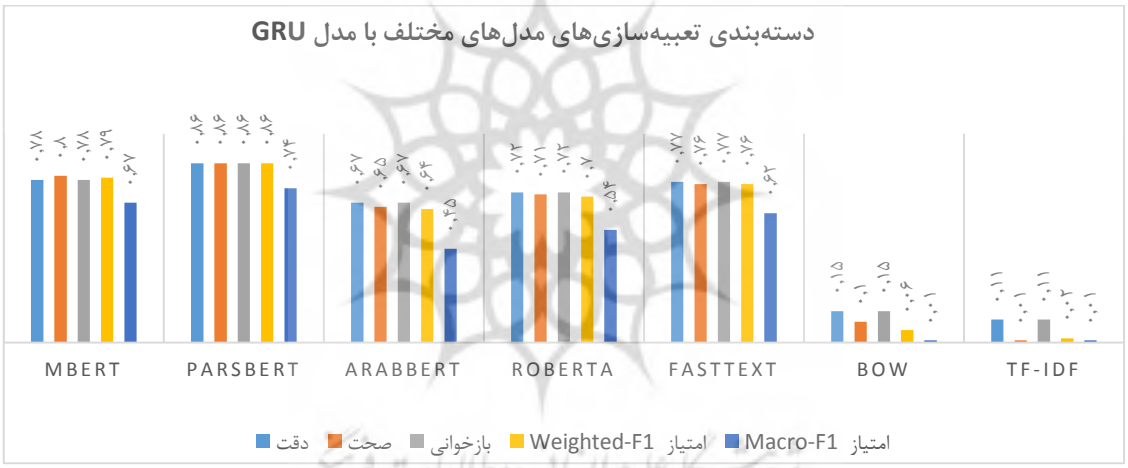
شکل (۹) عملکرد مدل RoBERTa را در دسته‌بندی تعبیه‌سازی‌های مدل‌های مختلف نمایش می‌دهد. نتایج نشان می‌دهد مدل RoBERTa در ترکیب با تعبیه‌سازی مدل‌های mBERT و ParsBERT و دسته‌بندی آن‌ها، عملکرد بهتری در مقایسه با خودش به‌تنهایی دارد.

شکل (۱۰) عملکرد مدل GRU را در دسته‌بندی تعبیه‌سازی‌های مدل‌های مختلف نمایش می‌دهد. نتایج نشان می‌دهد

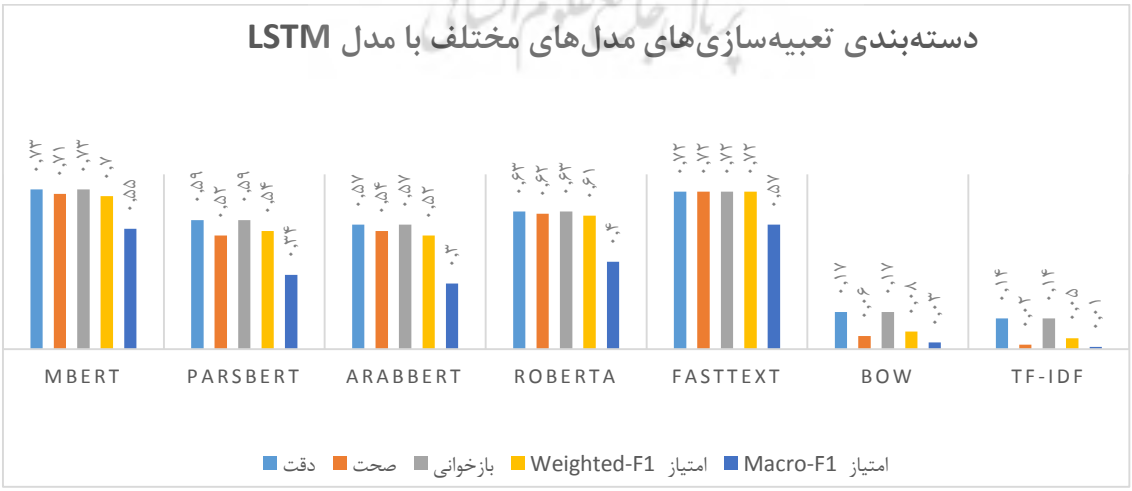
که مدل GRU در دسته‌بندی تعبیه‌سازی‌های پیشرفته ParsBERT و mBERT برتری محسوسی نسبت به سایر مدل‌ها دارد. این مدل در دسته‌بندی تعبیه‌های برداری مانند Bag of Words و TF-IDF عملکرد بسیار ضعیفی دارد، که نشان‌دهنده اهمیت استفاده از تعبیه‌های پیشرفته‌تر است.



شکل ۹. نمودار دسته‌بندی تعبیه‌سازی‌های مدل‌های مختلف با مدل RoBERTa



شکل ۱۰. نمودار دسته‌بندی تعبیه‌سازی‌های مدل‌های مختلف با مدل GRU

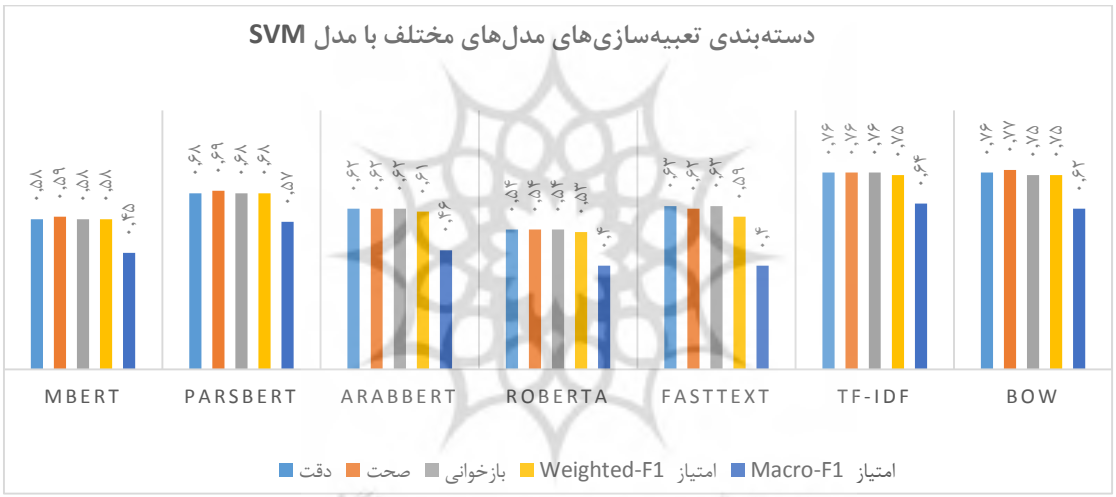


شکل ۱۱. نمودار دسته‌بندی تعبیه‌سازی‌های مدل‌های مختلف با مدل LSTM

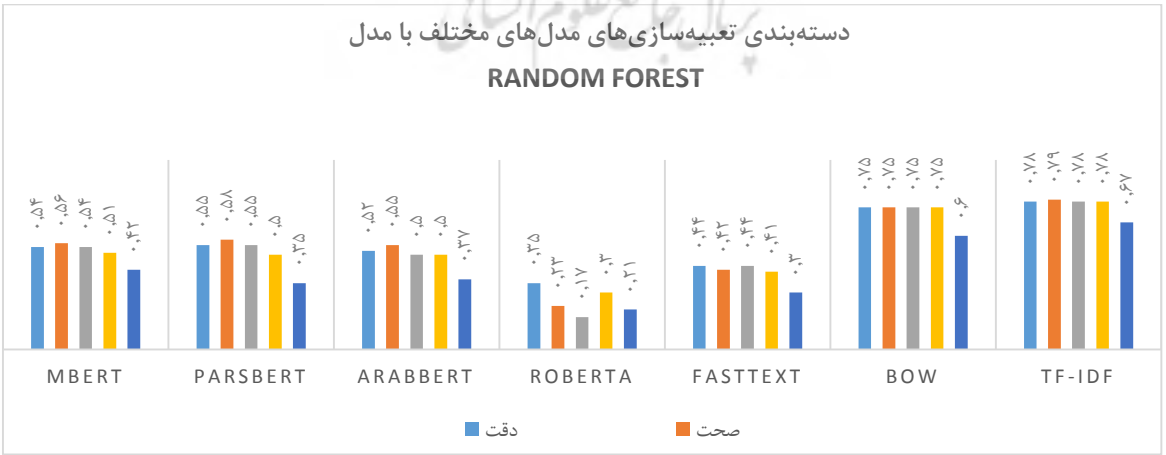
شکل (۱۱) عملکرد مدل LSTM را در دسته‌بندی تعبیه‌سازی‌های مدل‌های مختلف نمایش می‌دهد. نتایج نشان می‌دهد مدل LSTM در دسته‌بندی تعبیه‌سازی‌های مدل FastText عملکرد بهتری نسبت به سایرین دارد و همچنین نتایج آن بسیار نزدیک به عملکرد این مدل در دسته‌بندی تعبیه‌سازی‌های مدل mBERT است.

تحلیل نمودارهای Macro-F1 و Weighted-F1 نشان می‌دهد که مدل‌های شبکهٔ عصبی بازگشتی مانند GRU و LSTM نسبت به سایر دسته‌بندها، تفاوت بیشتری میان این دو امتیاز دارند. این اختلاف، حاکی از حساسیت بالاتر این مدل‌ها به نامتوازن بودن داده‌ها است. در واقع، عملکرد آن‌ها در مواجهه با کلاس‌های کم‌نمونه افت بیشتری دارد و این ویژگی باید در تحلیل نتایج آن‌ها مد نظر قرار گیرد.

شکل (۱۲) عملکرد مدل SVM را در دسته‌بندی تعبیه‌سازی‌های مدل‌های مختلف نمایش می‌دهد. نتایج نشان می‌دهد مدل SVM در دسته‌بندی تعبیه‌های سنتی مانند Bag of Words و TF-IDF عملکرد بهتری نسبت به سایرین دارد؛ ولی تعبیه‌های پیشرفته‌تر نیز نتایج نسبتاً مطلوبی ارائه می‌دهد.



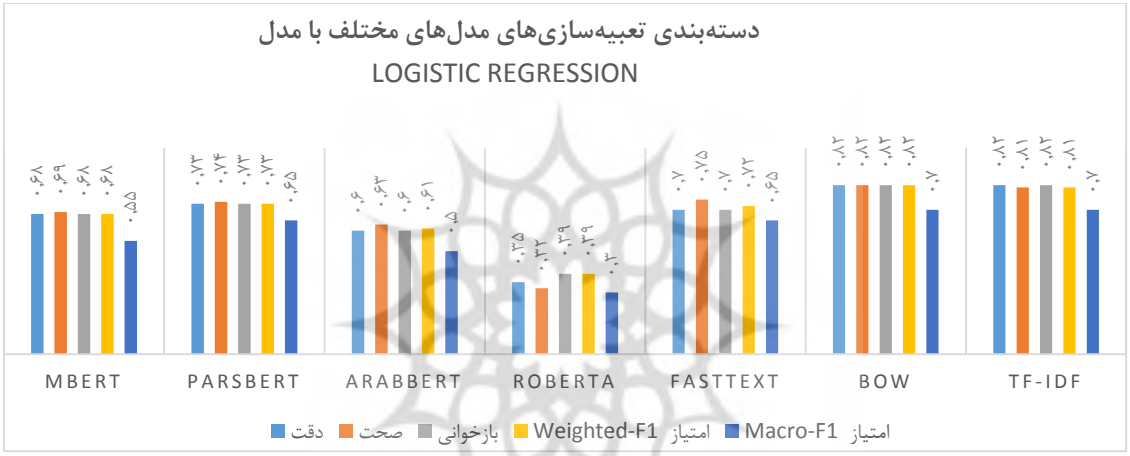
شکل ۱۲. نمودار دسته‌بندی تعبیه‌سازی‌های مدل‌های مختلف با مدل SVM



شکل ۱۳. نمودار دسته‌بندی تعبیه‌سازی‌های مدل‌های مختلف با مدل Random Forest

شکل (۱۳) عملکرد مدل Random Forest را در دسته‌بندی تعبیه‌سازی‌های مدل‌های مختلف نمایش می‌دهد. نتایج نشان می‌دهد مدل Random Forest در دسته‌بندی تعبیه‌های سنتی مانند Bag of Words و TF-IDF عملکرد بهتری نسبت به سایرین دارد. اگرچه تعبیه‌های پیشرفته‌تر نیز قادر به تولید نتایج پذیرفته‌شده‌تری بودند؛ بااین‌حال، عملکرد Random Forestدر ترکیب با برخی از روش‌های تعبیه‌سازی، نوسانات چشمگیری داشت که نشان می‌دهد این مدل نسبت به سایر مدل‌ها، حساسیت بیشتری نسبت به تعادل داده‌های آموزشی از خود بروز داده است.

شکل (۱۴) عملکرد مدل Logistic Regression را در دسته‌بندی تعبیه‌سازی‌های مدل‌های مختلف نمایش می‌دهد. نتایج نشان می‌دهد مدل Logistic Regression در دسته‌بندی تعبیه‌های سنتی مانند Bag of Words عملکرد بهتری نسبت به سایرین دارد؛ ولی این مدل در دسته‌بندی تعبیه‌های پیشرفته‌تر مانند ParsBERT و mBERT و FastText نیز نتایج بسیار خوبی ارائه می‌دهد.



شکل ۱۴. نمودار دسته‌بندی تعبیه‌سازی‌های مدل‌های مختلف با مدل Logistic Regression

برای ارزیابی معناداری تفاوت عملکرد مدل‌های ترکیبی در مقایسه با مدل‌های پایه، از آزمون آماری T-test زوجی استفاده شد. نتایج این آزمون حاکی از آن است که ترکیب مدل‌های ترنسفورمری با برخی از روش‌های تعبیه‌سازی، معماری‌های شبکه‌های عصبی و الگوریتم‌های یادگیری ماشین کلاسیک، در بسیاری از موارد به بهبود معنادار عملکرد طبقه‌بندی منجر شده‌اند (p-value < 0.05).

ترکیب مدل ParsBERT با FastText موجب بهبود معنادار عملکرد نسبت به مدل پایه شد (t=-2.7457, p=0.0335). همچنین، ترکیب ParsBERT با مدل‌های دیگری مانند mBERT (t=9.12, p<0.0001) ، AraBERT (t=3.58, p=0.0117)، LSTM (t=7.49, p=0.0003) و الگوریتم‌های کلاسیک یادگیری ماشین مانند: (t=8.00, p=0.0002) Random Forest و SVM (t=5.25, p=0.0019) نیز نتایج معناداری را به همراه داشت.

در میان ترکیب‌های مختلف mBERT، ترکیب آن با ParsBERT (t=12.85, p<0.0001)، Random Forest (t=6.47, p=0.0006) و FastText (t=-3.1692, p=0.0193) نیز نسبت به استفاده تنها از mBERT بهبودهای معناداری نشان دادند. درخصوص AraBERT، ترکیب آن با FastText بیشترین بهبود معنادار را نسبت به مدل پایه AraBERT ایجاد کرد (t=11.04, p<0.0001) .

افزون‌بر این، ترکیب این مدل با مدل‌های شبکه عصبی بازگشتی مانند LSTM (t=8.51, p<0.0001) و (t=4.52, p=0.0040) GRU والگوریتم‌های کلاسیک یادگیری ماشین نظیرRandom Forest (t=8.49, p<0.0001) و Logistic Regression (t=4.10, p=0.0063)، عملکرد بهتری نسبت به مدل پایه نشان داد.

اگرچه بیشتر ترکیب‌های مبتنی‌بر RoBERTa تفاوت معناداری نسبت به مدل پایه نداشتند، ترکیب RoBERTa با Logistic Regression (t=4.10, p=0.0063) عملکرد بهتری نسبت به استفاده از RoBERTa به‌تنهایی داشت.

در مجموع، نتایج نشان می‌دهند که ترکیب مدل‌های ترنسفورمری با مدل تعبیه‌سازی FastText، الگوریتم‌های کلاسیک یادگیری ماشین (مانند Random Forest و Logistic Regression) و همچنین مدل‌های شبکهٔ عصبی بازگشتی (مانند: LSTM و GRU)، به‌ویژه برای مدل‌هایی مانند AraBERT، ParsBERT وmBERT ، تأثیر چشمگیری در بهبود دقت و کارایی طبقه‌بندی متون فارسی دارند. افزون‌بر این، ترکیب‌هایی که از مدل‌های ناهمگون[1] تشکیل شده‌اند، یعنی مدل‌هایی با نقش‌ها و ساختارهای متفاوت، مانند ترکیب یک مدل ترنسفورمری با یک روش تعبیه‌سازی آماری نظیر FastText، یا با یک الگوریتم کلاسیک یادگیری ماشین مانند Logistic Regression، و همچنین با دسته‌بندهایی از نوع شبکه‌های عصبی بازگشتی مانند LSTM و GRU، در مقایسه با ترکیب‌های هم‌نوع مانند استفاده هم‌زمان از دو مدل ترنسفورمری، در بسیاری از موارد عملکرد بهتری از خود نشان داده‌اند. این نتایج بیانگر آن است که بهره‌گیری از ظرفیت‌های مکمل مدل‌های ناهمگون می‌تواند اثربخشی سیستم طبقه‌بندی را به‌طور معناداری افزایش دهد.

٦. بحث و بررسی

در این بخش به تحلیل نتایج حاصل از ارزیابی ترکیب‌های مختلف مدل‌های زبانی، روش‌های تعبیه‌سازی و الگوریتم‌های طبقه‌بندی پرداخته می‌شود. هدف اصلی این تحلیل، بررسی اثربخشی رویکردهای ترکیبی در طبقه‌بندی مفهومی و موضوعی نثرهای ادبی فارسی و مقایسهٔ عملکرد آن‌ها با مدل‌های منفرد است.

تحلیل نمودارهای مربوط به معیارهای ارزیابی در بخش پنجم شامل دقت، صحت، بازخوانی و امتیاز F1 (در دو حالت Macro و Weighted)، همراه با نتایج حاصل از آزمون آماری T زوجی، نشان می‌دهد که بسیاری از ترکیب‌های استفاده‌شده، به‌ویژه در تعامل میان مدل‌های زبانی مبتنی‌بر ترنسفورمر، تعبیه‌سازی‌های زمینه‌محور و مدل‌های شبکه‌های عصبی بازگشتی، توانسته‌اند عملکردی به‌مراتب بهتر و ازنظر آماری معنادار نسبت به روش‌های منفرد از خود نشان دهند. در ادامه، عملکرد سه گروه اصلی از ترکیب‌های بررسی‌شده شامل مدل‌های مبتنی‌بر BERT، شبکه‌های عصبی بازگشتی و الگوریتم‌های یادگیری ماشین، به‌صورت جداگانه تحلیل و تفسیر می‌شود.

٦ـ١. بررسی عملکرد مدل‌های BERT و مشتقات آن در دسته‌بندی و تعبیه‌سازی

مدل‌های زبانی مبتنی‌بر معماری ترنسفورمری مانند mBERT، ParsBERT، RoBERTa و AraBERT، در ترکیب با تعبیه‌سازی‌های مختلف، تفاوت‌های معناداری در عملکرد از خود نشان دادند. بررسی نمودارهای عملکرد مدل‌ها نشان می‌دهد که ترکیب مدل‌های ترنسفومری با استفاده از تعبیه‌سازی‌های پیشرفته‌ای مانند FastText و BERT، در مقایسه با

[1]. heterogeneous

مدل‌های مستقل، عملکرد بسیار بهتری از خود نشان دادند. این بهبود نه‌تنها در عملکرد کلی مدل‌ها مشهود است، بلکه به‌طور خاص به ارتقای چشمگیر معیارهای ارزیابی مختلف ازجمله دقت، بازخوانی و امتیاز F1 منجر شده است.

مدل ParsBERT، باتوجه‌به بهینه‌سازی‌های اختصاصی برای زبان فارسی، در اغلب ترکیب‌ها عملکرد بالاتری نسبت به سایر مدل‌ها از خود نشان داد و در ترکیب با FastText و دسته‌بندهایی مانند Logistic Regression یا GRU، بهبود معناداری نسبت به حالت استفادهٔ مستقل از ParsBERT داشت. این نتایج نشان می‌دهد که تلفیق این مدل با تعبیه‌سازی‌های بافت‌محور، بهبود معناداری در عملکرد معنایی آن ایجاد کرده است. به‌طور مشابه، mBERT نیز در ترکیب با FastText و LSTM عملکرد مناسبی داشته و ازنظر معیار امتیاز F1 نسبت به حالت پایه بهبود چشمگیری نشان داد. در مقابل، مدل ArabBERT، که عمدتاً برای متون عربی طراحی و بهینه شده است، در مقایسه با mBERT و ParsBERT عملکرد ضعیف‌تری را در پردازش متون فارسی نشان داد.

درمجموع، نتایج به‌دست‌آمده تأکید می‌کنند که ترکیب مدل‌های ترنسفورمری با تعبیه‌سازی‌های زمینه‌محور و دسته‌بندهای مناسب می‌تواند موجب هم‌افزایی میان ظرفیت‌های زبانی و آماری شده و به‌شکل معناداری عملکرد طبقه‌بندی متون ادبی فارسی را ارتقا دهد.

٦ـ۲. عملکرد مدل‌های شبکه عصبی بازگشتی (GRU و LSTM)

نتایج حاصل از ارزیابی مدل‌های شبکهٔ عصبی بازگشتی مانند GRU و LSTM نشان می‌دهد که عملکرد این مدل‌های دسته‌بندی به‌طور بسیاری تحت‌تأثیر نوع تعبیه‌سازی مورداستفاده قرار دارد. این مدل‌ها، به‌ویژه در ترکیب با تعبیه‌سازی‌هایی که قادر به حفظ ویژگی‌های معنایی و ساختاری زبان هستند، در استخراج روابط زبانی پیچیده و حفظ وابستگی‌های معنایی بلندمدت در متون ادبی، عملکرد مطلوبی از خود نشان دادند.

مدل GRU در ترکیب با تعبیه‌های مبتنی‌بر ترنسفورمر همچون ParsBERT، mBERT و RoBERTa، و همچنین در کنار FastText، عملکرد مطلوبی از خود نشان داد. ازسوی‌دیگر، مدل LSTM نیز در ترکیب با FastText و mBERT عملکرد خوبی داشت. این ترکیب‌ها نشان می‌دهند که مدل‌های شبکهٔ عصبی بازگشتی، در کنار تعبیه‌سازی‌هایی که توانایی حفظ اطلاعات زمینه‌ای دارند، می‌توانند در طبقه‌بندی متون ادبی کلاسیک فارسی بسیار مؤثر عمل کنند.

٦ـ۳. مقایسه عملکرد مدل‌های یادگیری ماشین نظارت‌شده (SVM، Random Forest، Logistic Regression)

مدل‌های یادگیری ماشین نظارت‌شده مانند SVM، Random Forest و Logistic Regression، ازنظر ساختار ساده‌تر از مدل‌های یادگیری عمیق هستند؛ اما در برخی ترکیب‌ها توانسته‌اند نتایج پذیرفته‌شده‌ای ارائه دهند. در ترکیب با تعبیه‌سازی‌های آماری مانند TF-IDF و Bag of Words، این مدل‌ها عملکرد مناسبی به‌ویژه در معیار دقت داشتند که نشان‌دهندهٔ تطابق آن‌ها با بردارهای ویژگی متکی بر فراوانی واژگان است.

مدل Logistic Regression، افزون‌بر عملکرد موفق در ترکیب با تعبیه‌های Bag of Words و FastText، توانست در ترکیب با ParsBERT به امتیاز بالای F1 دست یابد. این نتیجه بیانگر آن است که حتی دسته‌بندهای خطی نیز قادر به بهره‌برداری از مزایای تعبیه‌سازی‌های پیشرفته مانند ParsBERT هستند.

در مجموع، این مدل‌های یادگیری ماشین نظارت‌شده در برخی موارد نتایج مناسبی ارائه کردند؛ اما در مقایسه با مدل‌های پیشرفته‌تر نظیر مدل‌های مبتنی‌بر یادگیری عمیق و BERT، عملکرد پایین‌تری داشتند. این تفاوت عمدتاً به‌دلیل محدودیت مدل‌های خطی در درک وابستگی‌های معنایی پیچیده در متون فارسی است که توسط مدل‌های ترانسفورمری بهتر مدیریت می‌شود.

۷. نتیجه‌گیری

در این پژوهش، یک رویکرد ترکیبی نوآورانه برای طبقه‌بندی مفهومی و موضوعی نثرهای ادبی فارسی ارائه شد که با تلفیق مدل‌های زبانی مبتنی‌بر معماری ترنسفورمری، تکنیک‌های متنوع تعبیه‌سازی و الگوریتم‌های دسته‌بندی، در راستای رفع چالش‌های زبانی و معنایی متون ادبی کلاسیک عمل می‌کند. نتایج ارزیابی‌های انجام‌شده با استفاده از مجموعه‌ای از معیارهای رایج شامل دقت، صحت، بازخوانی و امتیاز F1 (در دو حالت Macro و Weighted) و همچنین آزمون آماری T زوجی، نشان داد که بسیاری از ترکیب‌های پیشنهادی، عملکرد مطلوبی از خود نشان دادند و در موارد متعددی، نسبت به مدل‌های منفرد بهبود معناداری را به همراه داشتند. این نتایج به‌وضوح اثربخشی رویکرد ترکیبی را در طبقه‌بندی متون ادبی فارسی تأیید می‌کند.

نتایج این پژوهش نشان می‌دهد که ترکیب مدل‌های زبانی با تعبیه‌سازی‌های زمینه‌محور نظیر FastText و استفادهٔ هم‌زمان از دسته‌بندهای پیشرفته شامل شبکه‌های عصبی بازگشتی و مدل‌های برداری، به‌گونه‌ای مؤثر توانسته است مزایای روش‌های آماری و یادگیری عمیق را با هم تلفیق کند. تلفیق تعبیه‌سازی‌های مدل‌های ترانسفورمری با این دسته‌بندها، نقاط قوت هر دو بخش را به‌خوبی تکمیل کرده است. این رویکرد ترکیبی نه‌تنها امکان استخراج ویژگی‌های زبانی دقیق‌تر را فراهم می‌سازد، بلکه با تقویت قابلیت‌های تشخیصی، به بهبود عملکرد کلی در طبقه‌بندی متون ادبی منجر شده و نتایج مطلوبی در پی داشته است. ازاین‌رو، دستیابی به بهترین نتایج در طبقه‌بندی نثرهای ادبی فارسی، نیازمند برقراری توازنی مناسب میان پیچیدگی مدل‌ها، روش‌های تعبیه‌سازی، نیازهای کاربردی و محدودیت منابع است.

فراتر از دستاوردهای فنی، این پژوهش با به‌کارگیری رویکردهای ترکیبی در تحلیل متون کلاسیک، امکان درک بهتر روابط معنایی را فراهم کرده و شرایط لازم برای تحلیل‌های دقیق‌تر را مهیا می‌سازد. این رویکرد همچنین به پژوهشگران کمک می‌کند تا با بهره‌گیری از روش‌های نوین، پیچیدگی‌های محتوایی این متون را بهتر تفسیر کنند. بدین‌ترتیب، پژوهش حاضر با ایجاد پیوند میان هوش مصنوعی و ادبیات، نه‌تنها به غنای مطالعات میان‌رشته‌ای در حوزهٔ ادب عرفانی می‌افزاید، بلکه زمینه را برای ایجاد سیستم‌های پیشرفتهٔ بازیابی اطلاعات ادبی و تحلیل متون کلاسیک هموار می‌سازد. در آینده، گسترش مجموعه‌های دادهٔ تخصصی‌تر در این حوزه می‌تواند ظرفیت‌های این رویکرد را تقویت کرده و دامنهٔ کاربردهای آن را گسترش دهد.

منابع

فیضی‌درخشی، محمدرضا، متقی‌نیا، زینب، و عسگری چناقلو، میثم (۱۴۰۱). طبقه‌بندی متون فارسی مبتنی‌بر شبکه‌های عصبی عمیق. *محاسبات نرم*، *۱۱*(۱)، ۱۲۰-۱۳۹. https://doi.org/10.22052/scj.2023.243182.1010

References

Ahmadi, P., Tabandeh, M., & Gholampour, I. (2016). Persian text classification based on topic models. In *Proceedings of the 2016 24th Iranian Conference on Electrical Engineering (ICEE)* (pp. 86–91). IEEE. https://doi.org/10.1109/iraniancee.2016.7585495

Antoun, W., Baly, F., & Hajj, H. (2020). Arabert: Transformer-based model for arabic language understanding. *arXiv preprint arXiv:2003.00104*. https://doi.org/10.48550/arXiv.2003.00104

Basiri, M. E., & Kabiri, A. (2017, April). Sentence-level sentiment analysis in Persian. In *Proceedings of the 2017 3rd international conference on pattern recognition and image analysis (IPRIA)* (pp. 84-89). IEEE. https://doi.org/10.1109/PRIA.2017.7983023

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

Cacciarelli, D., & Kulahci, M. (2024). Active learning for data streams: a survey. *Machine Learning*, *113*(1), 185-239. https://doi.org/10.1007/s10994-023-06454-2

Chicco, D., & Jurman, G. (2020). The advantages of the Matthews correlation coefficient (MCC) over F1 score and accuracy in binary classification evaluation. *BMC genomics*, *21*(1), 1-13. https://doi.org/10.1186/s12864-019-6413-7

Cho, K., Van Merriënboer, B., Gulcehre, C., Bahdanau, D., Bougares, F., Schwenk, H., & Bengio, Y. (2014). Learning phrase representations using RNN encoder-decoder for statistical machine translation. *arXiv preprint arXiv:1406.1078*. https://doi.org/10.48550/arXiv.1406.1078

Cortes, C., & Vapnik, V. (1995). Support-vector networks. *Machine learning*, *20*(3), 273-297. https://doi.org/10.1007/BF00994018

De Vries, W., van Cranenburgh, A., Bisazza, A., Caselli, T., van Noord, G., & Nissim, M. (2019). Bertje: A dutch bert model. *arXiv preprint arXiv:1912.09582*. https://doi.org/10.48550/arXiv.1912.09582

Devlin, J., Chang, M. W., Lee, K., & Toutanova, K. (2019, June). Bert: Pre-training of deep bidirectional transformers for language understanding. In *Proceedings of the 2019 conference of the North American chapter of the association for computational linguistics: human language technologies, volume 1 (long and short papers)* (pp. 4171-4186). https://doi.org/10.18653/v1/N19-1423

Farahani, M., Gharachorloo, M., Farahani, M., & Manthouri, M. (2021). Parsbert: Transformer-based model for persian language understanding. *Neural Processing Letters*, *53*(6), 3831-3847. https://doi.org/10.1007/s11063-021-10528-4

Farhoodi, Mojgan, & Yari, Alireza. (2010). *Applying machine learning algorithms for automatic Persian text classification.* In *Proceedings of the 2010 6th International Conference on Advanced Information Management and Service (IMS)* (pp. 318–323). IEEE. https://ieeexplore.ieee.org/abstract/document/5713467

Feizi-Derakhshi, M., Mottaghinia, Z., & Asgari-Chenaghlu, M. (2022). Persian text classification based on deep neural networks. *Soft Computing Journal*, *11*(1), 120–139 . 10.22052/scj.2023.243182.1010 [In Persian]

Hochreiter, S., & Schmidhuber, J. (1997). Long short-term memory. *Neural computation*, *9*(8), 1735-1780. https://doi.org/10.1162/neco.1997.9.8.1735

Hosmer Jr, D. W., Lemeshow, S., & Sturdivant, R. X. (2013). *Applied logistic regression*. John Wiley & Sons.

Howard, J., & Ruder, S. (2018). Universal language model fine-tuning for text classification. *arXiv preprint arXiv:1801.06146*. https://doi.org/10.48550/arXiv.1801.06146

Joulin, A., Grave, E., Bojanowski, P., & Mikolov, T. (2016). Bag of tricks for efficient text classification. *arXiv preprint arXiv:1607.01759*. https://doi.org/10.48550/arXiv.1607.01759

Karimi, S., & Shahrabadi, F. S. (2019). Sentiment analysis using BERT (pre-training language representations) and Deep Learning on Persian texts. *Technol. Deep Learn.* https://iust-deep-learning.github.io/972/static_files/project_reports/sent.pdf

Liu, Y., Ott, M., Goyal, N., Du, J., Joshi, M., Chen, D., ... & Stoyanov, V. (2019). Roberta: A robustly optimized bert pretraining approach. *arXiv preprint arXiv:1907.11692*. https://doi.org/10.48550/arXiv.1907.11692

Manning, C. D, Raghavan, P., & Schütze, H. (2008). Boolean retrieval. *Introduction to information retrieval*, 1-18.

Mo, Y., Qin, H., Dong, Y., Zhu, Z., & Li, Z. (2024). Large language model (llm) ai text generation detection based on transformer deep learning algorithm. *arXiv preprint arXiv:2405.06652*. https://doi.org/10.48550/arXiv.2405.06652

Opitz, J., & Burst, S. (2019). Macro f1 and macro f1. *arXiv preprint arXiv:1911.03347*. https://doi.org/10.48550/arXiv.1911.03347

Rumelhart, D. E., Hinton, G. E., & Williams, R. J. (1986). Learning representations by back-propagating errors. *nature*, *323*(6088), 533-536. https://doi.org/10.1038/323533a0

Xue, H., Huynh, D. Q., & Reynolds, M. (2018, March). SS-LSTM: A hierarchical LSTM model for pedestrian trajectory prediction. In *2018 IEEE winter conference on applications of computer vision (WACV)* (pp. 1186-1194). IEEE https://doi.org/10.1109/WACV.2018.00135