Niğde Ömer Halisdemir Üniversitesi Mühendislik Bilimleri Dergisi
Niğde Ömer Halisdemir University Journal of Engineering Sciences

Araştırma makalesi / Research article

www.dergipark.org.tr/tr/pub/ngumuh / www.dergipark.org.tr/en/pub/ngumuh

User interest classification on social media using machine learning and deep learning models: A multi-domain approach

Sosyal medyada makine öğrenmesi ve derin öğrenme tabanlı kullanıcı ilgi alanı sınıflandırması: Çok alanlı bir yaklaşım

Sultan Zeybek[1,*], Melih Kaçaman[2]

[1] *Istanbul University, Artificial Intelligence and Data Engineering, 34134, İstanbul, Türkiye*
[2] *Trendyol Group, Machine Learning Platform, 34485, İstanbul, Türkiye*

Abstract

The analysis of user-generated textual content provides valuable insights into user preferences in various domains, including politics, entertainment, health, sports, food, and technology. This study aims to automatically classify X user profiles based on interests using machine learning and deep learning algorithms. The objective is to categorize users into six interest areas with techniques including Naive Bayes, Logistic Regression, and Support Vector Machines, as well as LSTM, GRU, Bidirectional RNN, Conv1D, and Dense networks. Machine learning and deep learning models were compared using a pooled dataset, revealing that deep learning approaches generally improved generalization ability. Results indicate that while deep learning models achieve higher performance with large datasets, machine learning algorithms also perform competitively in certain categories. The findings highlight the potential of these models to support applications such as targeted content delivery, personalized recommendation systems, and user profiling on social media platforms.

Keywords: Machine learning, Social media, User profile analysis, Classification

Öz

Kullanıcı tarafından üretilen metin içeriklerinin analizi, siyaset, eğlence, sağlık, spor, yiyecek ve teknoloji gibi alanlarda kullanıcı tercihleri hakkında değerli bilgiler sunmaktadır. Bu çalışma, X kullanıcı profillerinin makine öğrenmesi ve derin öğrenme algoritmalarını kullanarak ilgi alanlarına göre otomatik sınıflandırmayı amaçlamaktadır. Araştırmanın temel hedefi, Naive Bayes, Lojistik Regresyon ve Destek Vektör Makineleri gibi makine öğrenmesi teknikleri ve LSTM, GRU, Bidirectional RNN, CNN ve Derin Sinir ağları gibi derin öğrenme modelleri kullanarak kullanıcıların ilgi alanlarının altı farklı kategoriye göre sınıflandırılmasını sağlamaktır. Makine öğrenmesi ve derin öğrenme modelleri, veri havuzlama yöntemi kullanılarak karşılaştırılmış ve derin öğrenme modellerinin genelleme yeteneğini artırmada daha etkili olduğu gözlemlenmiştir. Özellikle, derin öğrenme modellerinin büyük veri kümeleriyle daha iyi genelleme yapabildiği, ancak bazı kategorilerde makine öğrenmesi modellerinin de rekabetçi performans gösterdiği gözlemlenmiştir. Elde edilen sonuçlar, hedefe yönelik içerik sunumu, kişiselleştirilmiş öneri sistemleri ve sosyal medya platformlarında kullanıcı profillemesi gibi uygulama alanlarında önemli katkılar sağlama potansiyeline sahiptir.

Anahtar kelimeler: Makine öğrenmesi, Sosyal medya, Kullanıcı profili analizi, Sınıflandırma

1 Introduction

Social media platforms have become essential research tools, offering unprecedented access to real-time data. The utilisation of social media has significantly transformed the research landscape by enabling the exploration of new topics and enhancing the depth and validity of research findings in today's interconnected digital world. Among these platforms, Twitter (now X) has gained considerable popularity as a microblogging service, where users share concise, real-time updates, ranging from personal insights to breaking news. This unique characteristic makes X an invaluable data source for research purposes. Through analysing the diverse information shared on X, researchers can gain unique insights into individuals' lives, preferences, interests, and broader patterns of human behaviour, trends, and interactions.

Social media mining has emerged as a multidimensional field, encompassing various tasks [1] such as topic modelling [2], fake news detection [3], sentiment analysis [4], user profiling [5], threats and insults detection [6], recommendation systems [7], and others [8]. To perform these tasks effectively, substantial amounts of data must be collected. In this context, X serves as a platform for gathering multidimensional data, offering valuable insights into users' communication and sharing behaviours.

Personal shares on social media platforms hold great potential for determining users' preferences and social characteristics. It's essential to explore how user behaviours

* Sorumlu yazar / Corresponding author, e-posta / e-mail: sultan.zeybek@istanbul.edu.tr (S. Zeybek)
Geliş / Received: 12.03.2025 Kabul / Accepted: 03.09.2025 Yayımlanma / Published: xx.xx.20xx
doi: 10.28948/ngumuh.1655764

on platforms like Twitter (X) [9]. In this context, algorithms have been developed for multi-criteria recommender systems. These systems provide valuable insights into user behaviour and preferences, such as suggesting similar profiles on X. They are also crucial in market analyses, where they enhance business intelligence by generating profile suggestions based on user interactions, such as shares and likes. This strategic integration of user-generated data into business intelligence contributes to more informed decision-making processes.

The early studies on social media profile analysis are most likely based on traditional machine learning techniques, including Naive Bayes, Logistic Regression, and Support Vector Machines (SVM), that have been used to classify users based on their interests, behaviours, and sentiments extracted from textual data on social media platforms [10]. Abel et al. analysed user interactions and behaviours on Twitter for personalised news recommendations. The authors classified users in the topics they discussed by using machine learning techniques. In this way, they showed how important correct user interest modelling is for real-time systems [11]. Similar to user profile detection studies, machine learning techniques have also been applied in different contexts [12], that integrating linguistic and behavioral features not only improved overall accuracy but also achieved promising results for difficult cases, such as short or repetitive comments. This highlights the flexibility of machine learning models and their adaptability across diverse tasks, including profile classification, spam detection, and recommendation systems.

This study aims to analyse X profiles across various interest areas including politics, entertainment, health, sports, food, and technology using machine learning algorithms. The textual content of tweets is utilised to detect profile preferences according to these categories. To achieve this, machine learning algorithms such as Naive Bayes, Logistic Regression, and SVM have been employed for classification tasks.

In the second part of the study, the machine learning and deep learning methods used in the developed classification model are presented. Subsequently, the dataset and data preprocessing stages are discussed, and comparative performance results are reported. Finally, the study concludes with results and suggestions for future work.

2 Material and methods

This study utilised a dataset gathered from a popular social media platform Twitter (X), specifically focusing on user-generated content such as tweets. The dataset was collected using the platform’s API over a period of time for English tweets, and these tweets were further categorised into six distinct interest categories: politics, entertainment, health, sports, food, and technology. A labelling process was carried out by human annotators to ensure that each tweet was correctly categorised.

2.1 Dataset and pre-processing

The performance of classification models significantly depends on the quality and quantity of training data. While manual labelling is often used for creating training datasets, this approach is very time-consuming. Distant supervision has been used as an automatic labeling of data using external knowledge sources, such as hashtags or verified account categories, without manual annotation. However, recognising that hashtags alone do not comprehensively represent user interests and may introduce noise, the data collection approach was refined. Instead, tweets were gathered from 36 carefully selected and reputable content sources, including verified newspapers, magazines, and domain-specific accounts. Each interest category was represented by accounts known for their subject-matter expertise. For example, political content was collected from verified accounts such as @politico, @thehill, @foxnewspolitics, @ReutersPolitics, and @CNNPolitics. This approach ensured a more balanced and domain-relevant dataset, reducing bias and providing higher-quality textual data for model training.

Pre-processing plays a crucial role in preparing social media data for machine learning and deep learning models due to the noisy and unstructured nature of tweets. In this study, several preprocessing steps were applied to standardize the raw text data. First, URLs, user mentions, hashtags, special characters, numbers, and punctuation marks were removed to eliminate irrelevant noise. All text was converted to lowercase to ensure consistency during tokenization. Common stopwords, which carry little semantic meaning (e.g., "the", "is", "and"), were removed to focus the analysis on the most informative words. The cleaned text was then tokenized, converting sentences into sequences of words. For the deep learning models, a text vectorization layer was used to transform the tokens into integer sequences, and an embedding layer mapped these integers into dense vector representations. These preprocessing steps ensure the data is standardised, allowing for more effective training of the classification algorithms. For each category, 5,000 tweets were collected. To develop a binary classifier model for each category, an additional 5,000 tweets from the remaining categories were evenly distributed and combined with the primary category, resulting in datasets consisting of 10,000 tweets for each classification task. The following machine learning and deep learning methods were used for classification performance. Figure 1 represents the proposed pipeline of the machine learning algorithms for classification.

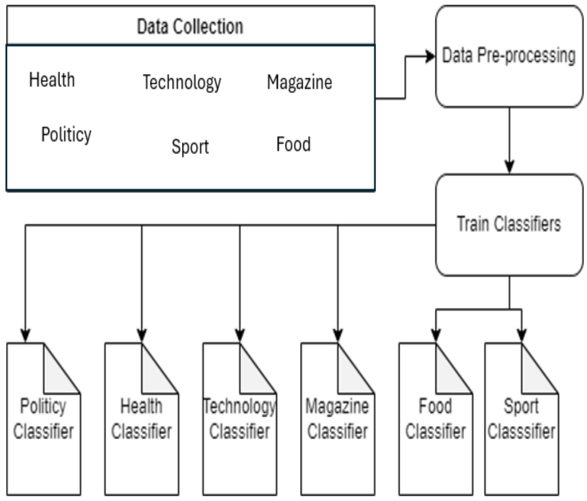

Figure 1. The pipeline of the proposed models.

Table 1 provides example tweets extracted from the dataset, illustrating the diversity of content across the six predefined interest domains. These samples illustrate the contextual diversity and linguistic variation of each category, representing the real-world complexity of social media content used to train the classification models.

Table 1. Domain-specific examples of the collected data.

Domain	Tweet
Technology	Google Cloud launches its new data loss prevention tool for BigQuery
Health-Fitness	Heart rate training is a great way to better understand how external factors such as sleep, hydration, and even caffeine are affecting your performance inside and outside of the gym.
Food	It's so easy you can whip up a batch before your morning coffee.
Politics	Each Friday, go behind the scenes with the women reshaping politics, policy and power in Washington and around the world.
Magazine	A Royal Reunion! Prince Harry and Meghan Markle visit with Queen Elizabeth II in London for the first time since they left their royal duties in early 2020.
Sport	Our NBA writers ranked their top 25 under 25.

2.2 Machine learning methods

2.2.1 Support Vector Machines

SVM is a powerful algorithm commonly used in the field of machine learning to solve classification problems. This algorithm focuses on finding the optimal decision boundaries between classes when determining to which class a data point belongs [13]. SVM defines the decision boundaries by maximising the gap between the classes. These decision boundaries are drawn based on the data points that are most critical between the classes, called 'support vectors.' The algorithm represents the data points in a vector space and finds the best-separating line in this space. This line is chosen in such a way as to maximise the margin between the classes. SVM can also apply kernel functions, such as linear, polynomial, or radial basis function (RBF) kernels, to transform the data into higher dimensions for more complex classifications. An example of linearly non-separable datasets is illustrated in Figure 1.

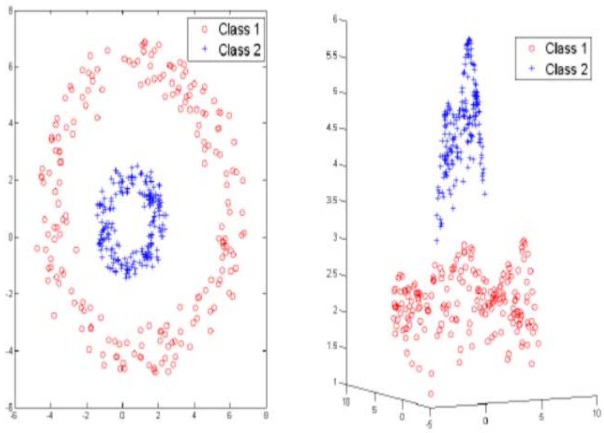

Figure 2. SVM for linearly non-separable datasets.

2.2.2 Naïve Bayes

Naïve Bayes is a probabilistic classifier based on Bayes' Theorem, which performs well in various domains such as text classification and spam detection due to its efficiency and ability to handle high-dimensional data.

$$\mathrm{P}(y \mid X) = \frac{\mathrm{P}\ (X \mid y)\ P(y)}{P(X)} \tag{1}$$

Equation (1) is the key formula used in the Naïve Bayes classifier, where $\mathrm{P}\ (y|X)$ is the posterior probability that we want to calculate to make the classification decision. $\mathrm{P}(X|y)$ the prprobability of observing the feature set X given the class is y, $P(y)$ is the probability of class y occurring before considering the features, and $P(X)$ is the total probability of observing the feature set X across all possible classes. The Naive Bayes classifier computes the posterior probability where $\mathrm{P}\ (y|X)$ for each class y and then assigns the class label with the highest posterior probability to the input sample.

2.2.3 Logistic Regression

Logistic Regression is a widely used machine learning algorithm that applies a logistic (sigmoid) function, and can be used for both binary and multi-class classification tasks. In this study, the logistic regression classifier was employed to predict the probability that an input belongs to one of two possible classes across six different categories, using the following formula:

$$\mathrm{P}(y = 1 \mid X) = \frac{1}{1 + e^{-z}} = \sigma\ (z) \tag{2}$$

Here, $P(y = 1| X)$ represents the probability that the class label is 1 given the input features X, z is the linear combination of input features, and $\sigma(z)$ is the sigmoid function. For a binary classification task, the threshold is defined as 0.5 to predict the class. If $P(y = 1| X) > 0.5$ the model predicts class as 1, otherwise the prediction is 0.

2.3 Deep learning methods

Deep learning is a subfield of machine learning that focuses on modelling complex data patterns using multi-layered artificial neural networks. This method offers a hierarchical feature learning process, where low-level features are used to learn high-level features. Thus, the model can understand data through a more comprehensive architecture. Due to its ability to automatically learn representations from raw data, deep learning methods often outperform traditional machine learning methods in many tasks. In this study, we proposed to compare various deep learning architectures, including deep neural networks (DNNs), recurrent neural networks (RNNs), and convolutional neural networks (CNNs).

The RNN model is designed to learn sequential temporal data with input variable $x^{(i)} = (x_1, x_2, \ldots, x_{t-1}, x_t)$ which represent a word embeddings vectors. The deep RNN model uses the following recurrence formula:

$$\begin{aligned} h_t &= \tan h(W_{hh}h_{t-1} + W_{xh}x_t + b_h) \\ y &= sigmoi\,d(W_{hy}h_t + b_y) \end{aligned} \tag{3}$$

which the hidden state h_t carries information from the previous time step h_{t-1} and employs it to classify the given observation $x^{(i)}$. The learning process involves updating shared learnable weight parameters W_{hh}, W_{xh}, W_{hy}, along with learnable bias terms b_h and b_y.

Long Short-Term Memory (LSTM) Networks represent a specialized variant of recurrent language models [14], specifically designed to handle long-term dependencies within sequential data of varying lengths. These LSTM layers are structured across the sequence, enabling the transfer of information using the following equations with learnable parameters W_{ih}, U_{ih}, W_{hf}, U_{hf}, W_{ho}, U_{ho}, W_u, U_u, b^i, b^f, b^o, b^u:

$$\begin{aligned} ig_t &= \varphi(W_{ih}x_t + U_{ih}h_{t-1} + b^i) \\ fg_t &= \varphi\left(W_{hf}x_t + U_{hf}h_{t-1} + b^f\right) \\ og_t &= \varphi(W_{ho}x_t + U_{ho}h_{t-1} + b^o) \\ ug_t &= \psi(W_u x_t + U_u h_{t-1} + b^u) \\ mc_t &= ig_t \odot ug_t + fg_t \odot mc_{t-1} \\ hg_t &= og_t \odot \psi(mc_t) \end{aligned} \tag{4}$$

In these equations, the values of $ig_t, fg_t, og_t \in \{0, 1\}$. The learning process occurs through sequential computational steps using the sigmoid function denoted as φ,

Gated Recurrent Units (GRUs) present an alternative RNN architecture designed to address the training time inefficiencies observed in standard RNNs. The GRU architecture integrates reset and update gates, enabling these gates to learn information akin to the LSTM memory unit, utilizing the following equations:

$$\begin{aligned} z_t &= \varphi(U_z x_t + W_z hg_{t-1}) \\ rg_t &= \varphi(U_r x_t + W_r hg_{t-1}) \\ s_t &= \psi(U_z x_t + W_s hg_{t-1} rg_t) \\ hg_t &= (1 - z_t)s_t + z_t hg_{t-1} \end{aligned} \tag{5}$$

BERT (Bidirectional Encoder Representations from Transformers) is a model used in the field of natural language processing (NLP) and refers to the representation obtained from bidirectional transformers. Introduced by Google in 2018, BERT has made significant progress in various NLP tasks. The main innovation of BERT is that it can understand the context from both left and right directions thanks to the bidirectional training process. This feature allows the model to better understand the meaning of the words in the sentence. BERT's architecture is based on the Transformer model and uses attention mechanisms to learn text representation [15].

2.3.1 Model architectures and training configurations

To evaluate the effectiveness of machine learning and deep learning approaches, several models were implemented using the TensorFlow and scikit-learn libraries. All deep learning models employed an initial text preprocessing stage, where the tweets were vectorized using a TextVectorization layer with a maximum vocabulary size of 10,000 words and an output sequence length of 18 tokens. The text was then embedded into a dense vector space using an Embedding layer of dimension 128.

Table 2. Deep Learning models and their architectures.

Deep Learning Model	Model Architecture
Simple Dense Model	Embedding layer, followed by a GlobalAveragePooling1D and a Dense output layer with a softmax activation.
LSTM and GRU Models	Each used a single recurrent layer of 64 units with tanh activation, followed by a Dense output layer
Bidirectional LSTM	A single LSTM layer (64 units) in a bidirectional structure, enhancing its ability to capture dependencies from both directions in the text
1D CNN	The CNN model included two Conv1D layers, each with 32 filters and a kernel size of 5, followed by a GlobalMaxPooling1D layer and a Dense output layer.

Table 2 reports the model architectures and hyperparameters of the proposed deep learning models. After obtaining results on unpooled data, we pooled the data and trained the best-performing machine learning model on this pooled larger data in order to compare the performance with deep learning models. The pooled dataset is balanced, each category contains 5,000 tweets where all classes contribute equally to the model's training and evaluation process. The primary aim of this approach was to increase the size of the dataset, and the generalisation ability of deep learning models, and to avoid overfitting using 30,000 tweets for multi-class model. All deep learning models, including LSTM, GRU, Bidirectional RNN, Conv1D, and Dense networks, all optimized using the Adam optimizer with a learning rate of 0.001. Categorical cross-entropy was chosen as the cost function, and word embeddings were applied to all models. Regularized versions of LSTM and Bidirectional RNN incorporate L2 regularization and dropout layers with 0.4 to enhance generalization and prevent overfitting. The Conv1D model uses two convolutional layers with a 5x5 kernel and ReLU activation, making it suitable for learning local text features. Overall, these models were designed to evaluate different architectures for user interest classification in social media.

3 Results and discussion

This section presents the results and discussion obtained from the classification models. For the training of machine learning models, various algorithms were applied using the sci-kit-learn library, and hyperparameter optimisation was performed using Grid Search and Cross-Validation techniques. By comparing training and test accuracy, the risk of overfitting was minimized, and the results were evaluated. The classification accuracy obtained from the models' performance evaluations demonstrated the effectiveness of the machine learning algorithms. In the deep learning phase, models were employed using the Google Colab environment. The learning rate was adjusted to optimize the models' performance, and an early stopping strategy was used to further fine-tune the models.

3.1 Model performance metrics

The performance of the classification models was evaluated using various metrics. The performance criteria included accuracy, precision, recall, and F1 score, which are expressed by the following equations:

$$Accuracy = \frac{TP + TN}{(TP + FN) + (TN + FP)} \tag{6}$$

$$Precision = \frac{TP}{TP + FP} \tag{7}$$

$$Recall = \frac{TP}{TP + FN} \tag{8}$$

$$F1\,Score = 2 \text{ x } \frac{Precision \text{ x } Recall}{Precision + Recall} \tag{9}$$

Here, accuracy indicates how well the model makes correct predictions across the entire dataset and is calculated as the ratio of true positives (TP) and true negatives (TN) to the total number of predictions. Precision refers to the proportion of correctly predicted positive instances among all instances that the model predicted as positive. It is calculated as the ratio of true positives (TP) to the total positive predictions, including false positives (FP). Recall measures how accurately the model identifies actual positive instances; it is calculated as the ratio of true positives (TP) to the total actual positives, including false negatives (FN). The F1 score is the harmonic mean of precision and recall, balancing the two metrics to summarise the model's overall performance.

3.2 Experimental results

Comparative performance evaluations are presented based on the performance of the machine learning and deep learning models, respectively. As shown in Table 3, SVM model with a linear kernel achieved validation accuracy accuracies ranging from 0.8068 for health to 0.9230 for politics. However, the model's test accuracy varied more significantly across categories, with the highest performance in politics (0.7835) and sports (0.7621), and the lowest in technology (0.6042). The F1-Score values gave similar results. Table 4 reports the results for the model SVM with RBF Kernel. RBF Kernel performed better than the Linear Kernel, but SVM had difficulty generalising the categories. Table 5 reports the performance results for the Logistic Regression classifier. As can be seen from Table 5, Logistic Regression performed more balanced across categories but still has difficulties handling the complex nature of the non-linear data categories like technology. Table 6 reports the performance of the Naïve Bayes classifier with the highest validation accuracy overall. Test results are also better than the other machine learning algorithms, however, the F1-score, recall, and precision metrics performed non-balanced. The performance results of the machine learning algorithms showed that the results mostly depend on the characteristics of the data in each category. The Naive Bayes classifier achieved the highest validation and test accuracies in simpler categories like technology and food, but cannot handle the difficulties of the politics category. The performance of the SVM model depends on the kernel used. Figure 3 and Figure 4 represent the comparative results for different kernel metrics, SVM with RBF kernel generally outperformed the Linear Kernel across all categories. However, SVM with Linear kernel performed better in the Politics category, achieving a higher test F1-score and better recall.

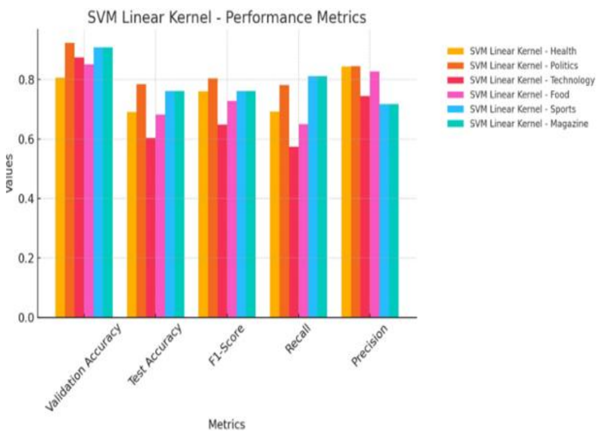

Figure 3. Performance metrics for SVM Linear Kernel.

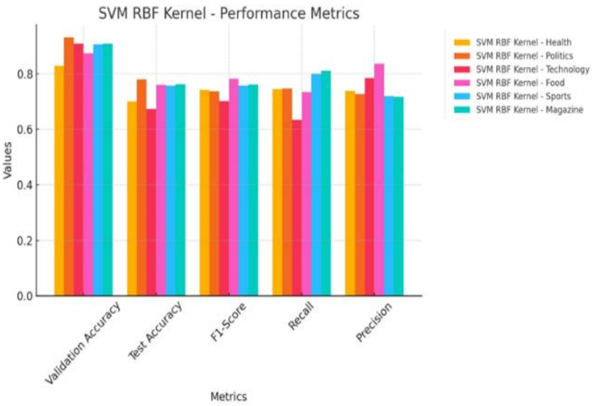

Figure 4. Performance metrics for SVM RBF Kernel.

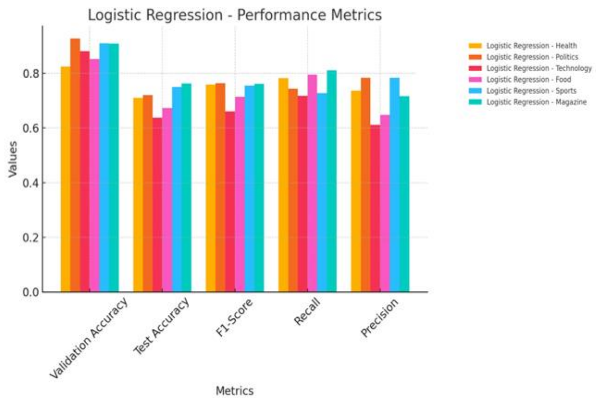

Figure 5. Performance metrics for LR.

Additionally, Figure 5 and Figure 6 present the performance results for the Logistic Regression and Naive Bayes classifiers, respectively. The Naive Bayes classifier achieved the highest accuracy in the technology and food categories, while Logistic Regression performed balanced across all categories. The performance evaluation results including validation and test accuracy, and also F1-score, recall and precision metrics are given in Table 3, Table 4, Table 5, and Table 6 for the SVM with Linear Kernel, SVM with RBF Kernel, Logistic Regression, and Naive Bayes, respectively.

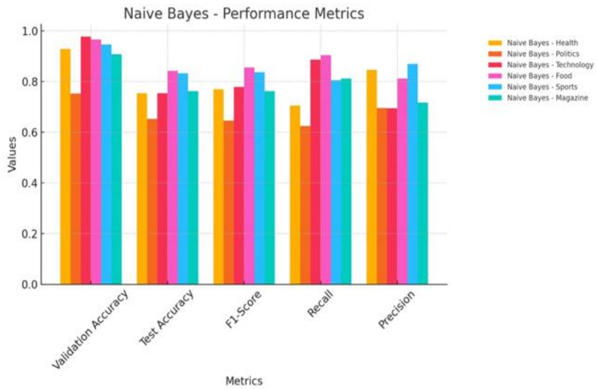

Figure 6. Performance metrics for NB.

Among all classifiers, the Technology category has been detected as the hardest to classify, showing the lowest performance across models, while the Food category achieved the highest classification success. Naive Bayes outperformed the other three classifiers, particularly in the more challenging categories such as Technology and Food. In addition, SVM with Linear kernel achieved the highest F1-score and recall in the Politics category, which makes it more suitable for linearly separable patterns like politics, as expected.

Table 3. Performance results for the SVM Linear Kernel.

Metrics	Health	Politics	Technology	Food	Spor	Magazine
Val. Acc.	0.8068	**0.9232**	0.8747	0.8507	0.9086	0.9086
Test Acc.	0.6912	**0.7835**	0.6042	0.6815	0.7621	0.7621
F1-Score	0.7607	**0.8042**	0.6480	0.7278	0.7613	0.7613
Recall	0.6923	0.7813	0.5741	0.6496	**0.8116**	**0.8116**
Precision	**0.8440**	**0.8440**	0.7438	0.8274	0.7169	0.7169

Table 4. Performance results for the SVM RBF Kernel.

Metrics	Health	Politics	Technology	Food	Spor	Magazine
Val. Acc.	0.8286	**0.9316**	0.9085	0.8742	0.9057	0.9086
Test Acc.	0.7009	**0.7793**	0.6730	0.7602	0.7573	0.7621
F1-Score	0.7417	0.7367	0.7017	**0.7820**	0.7573	0.7613
Recall	0.7447	0.7467	0.6343	0.7347	0.8000	**0.8116**
Precision	0.7386	0.7267	0.7851	**0.8357**	0.7190	0.7169

Table 5. Performance results for the Logistic Regression.

Metrics	Health	Politics	Technology	Food	Spor	Magazine
Val. Acc.	0.8247	**0.9275**	0.8812	0.8518	0.9093	0.9086
Test Acc.	0.7105	0.7205	0.6381	0.6725	0.7495	**0.7621**
F1-Score	0.7587	**0.7637**	0.6603	0.7140	0.7545	0.7613
Recall	0.7830	0.7437	0.7179	0.7944	0.7272	**0.8116**
Precision	0.7359	0.7837	0.6112	0.6483	**0.7839**	0.7169

Table 6. Performance results for the Naive Bayes classifier.

Metrics	Health	Politics	Technology	Food	Spor	Magazine
Val. Acc.	0.9295	0.7524	**0.9780**	0.9660	0.9466	0.9086
Test Acc.	0.7543	0.6524	0.7535	**0.8426**	0.8332	0.7621
F1-Score	0.7695	0.6456	0.7789	**0.8553**	0.8364	0.7613
Recall	0.7055	0.6253	0.8863	**0.9039**	0.8057	0.8116
Precision	0.8463	0.6956	0.6947	0.8116	**0.8695**	0.7169

Table 7. Comparative performance of the best machine learning model with deep learning models

Model	Acc.	Macro F1-Score	Macro Precision	Macro Recall
Simple Dense Model	**0.8968**	**0.8962**	**0.8968**	**0.8962**
LSTM Model	0.8739	0.8737	0.8754	0.8731
GRU Model	0.2175	0.1505	0.2050	0.2078
Bidirectional RNN (BiRNN) Model	0.8789	0.8776	0.8782	0.8770
Conv1D Model	0.8640	0.8635	0.8641	0.8631
Simple Dense Model with Regularization	0.8900	0.8895	0.8895	0.8893
LSTM Model with Regularization	0.8848	0.8837	0.8845	0.8838
BiRNN Model with Regularization	0.8811	0.8802	0.8805	0.8801
Naïve Bayes Model	0.8230	0.8217	0.8241	0.8225

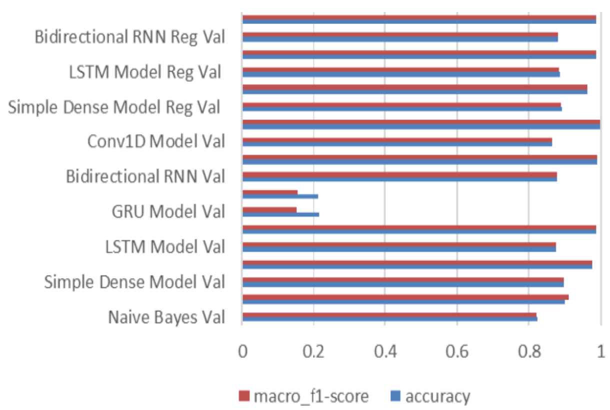

Figure 7. Performance metrics for deep learning models.

Table 7 and Figure 7 report the performance comparisons observed in the best machine learning and deep learning models on the validation pooled dataset. Since the pooled dataset contains 6 different categories the model performances have been compared based on the macro F1-score, which calculates the F1-score independently for each class and then takes the average. The balanced dataset is critical to avoid a significant risk of bias for the model towards one category of the model, and the macro F1-score provides a fair comparison of different classifiers for both machine learning and deep learning models, according to the experimental results, of the deep learning models as GRU did not perform as well as expected. One of the primary reasons for this is the short, irregular, and noisy nature of social media data, and the multi-class nature of the pooled dataset contains that contain 6-different categories.

4 Conclusions

This study aimed to classify user interest in social media using machine learning and deep learning models, focusing on six different categories including politics, entertainment, health, sports, food, and technology. Classification performance indicated that the Technology category was the most difficult to classify with generally lower performance in all of the models, and the Food category provided the highest classification rate. Among the machine learning classifiers, Naive Bayes performed better than the other three models particularly highest in the challenging Technology and Food categories. Meanwhile, SVM with a Linear kernel achieved the best F1-score and recall for the Politics category, more appropriate for linearly separable patterns such as political content, as expected. According to the comparative results, the Simple Dense Model performed with the highest macro-F1 score and accuracy among machine learning and deep learning models proved that model selection is a crucial problem depending on the target category. Limited dataset size, hyperparameter choices, and computational constraints cause lower performance metrics for some models like GRU. The integration of various models might be a practical approach to the enhancement of user interest classification on social media. For future works, advanced deep learning models and transformed-based models such as BERT will be evaluated for better performance in classifying the short, noisy, and highly dynamic nature of social media text. The hyper-parameter optimisation including using smaller batch size could also be with a multi-label classification approach could also be beneficial to improve user interest classification across different domains.

Conflict of Interest

The authors of this study declare that there is no conflict of interest regarding the conduct of the research, the evaluation of the results, or the publication process. The authors have no direct or indirect financial or personal gain from this study

Similarity Score (iThenticate): 10%.

References

[1] R. Zafarani, M. A. Abbasi, and H. Liu. Social media mining: an introduction. Cambridge University Press, 2014. https://doi.org/10.1017/CBO9781139088510.

[2] I. Himelboim, M. A. Smith, L. Rainie, B. Shneiderman, and C. Espina. Classifying Twitter Topic-Networks Using Social Network Analysis. Social Media + Society, 3(1), 2017. https://doi.org/10.1177/2056305117691545.

[3] M. Bozkuş, E. Arıcı, and S. Zeybek. Fake News Detection on Social Media Data using Community Notes with Machine Learning. 2025 9th International Symposium on Innovative Approaches in Smart Technologies (ISAS), Gaziantep, Turkiye, 1-6, 2025. https://doi.org/10.1109/ISAS66241.2025.11101905.

[4] S. Zeybek, E. Koç, and A. Seçer. MS-TR: A Morphologically enriched sentiment Treebank and recursive deep models for compositional semantics in Turkish. Cogent Engineering, 8(1), 1893621, 2021. https://doi.org/10.1080/23311916.2021.1893621.

[5] A. S. M. Alharbi, and E. Doncker. Twitter sentiment analysis with a deep neural network: An enhanced approach using user behavioral information. Cognitive Systems Research, 54, 50-61, 2019. https://doi.org/10.1016/j.cogsys.2018.10.001.

[6] S. Zeybek, B. Alkın, Y. Kaya. Derin öğrenme ve makine öğrenmesi yöntemleri ile sosyal medya verilerinden suç tespiti. NOHU J. Eng. Sci., vol. 14, no. 1, 175–182, 2025. https://doi.org/10.28948/ngumuh.1551734.

[7] S. Wang, X. Zhang, Y. Wang, and F. Ricci. Trustworthy recommender systems. ACM Transactions on Intelligent Systems and Technology, 15(4), 1-20, 2024. https://doi.org/10.1145/3627826.

[8] M. Salemdeeb and S. Sahmoud. What to predict from Twitter Data?. 3rd International Conference on Computing and Information Technology (ICCIT), Tabuk, Saudi Arabia, 237-242, 2023. https://doi.org/10.1109/ICCIT58132.2023.10273883.

[9] K. Khoirunurrofik, C. D. Endrina, and A. M. Zulkarnain. Exploring the public sentiment of local community on major infrastructure development: Evidence from media news and Twitter data. Journal of Human Behavior in the Social Environment, 34(3), 423-443, 2024. https://doi.org/10.1080/10911359.2023.2249962.

[10] N. N. Daud, S. H. Ab Hamid, M. Saadoon, F. Sahran, and N. B. Anuar. Applications of link prediction in social networks: A review. Journal of Network and Computer Applications, 166, 102716, 2020. https://doi.org/10.1016/j.jnca.2020.102716.

[11] F. Abel, Q. Gao, G. J. Houben, and K. Tao. Analyzing user modelling on Twitter for personalized news recommendations. Proceedings of the 19th ACM International Conference on User Modeling, Adaption, and Personalization, 1-12, 2011. https://doi.org/10.1007/978-3-642-22362-4_1.

[12] S. S. Atali, B. Kozlucali, and S. Zeybek. Detecting Spam Comments in Product Reviews on E-Commerce Websites. 2025 9th International Symposium on Innovative Approaches in Smart Technologies (ISAS), Gaziantep, Turkiye, 1-4, 2025. https://doi.org/10.1109/ISAS66241.2025.11101848.

[13] C. Cortes, V. Vapnik. Support-vector networks. *Mach Learn* 20, 273–297, 1995. https://doi.org/10.1023/A:1022627411411.

[14] S. Hochreiter and J. Schmidhuber. Long short-term memory. Neural Computation, 9(8), 1735-1780, 1997. https://doi.org/10.1162/neco.1997.9.8.1735.

[15] J. Devlin, M. W. Chang, K. Lee, & K. Toutanova. BERT: Pre-training of Deep Bidirectional Transformers for Language Understanding. Proceedings of NAACL-HLT, 2019. https://doi.org/10.18653/v1/N19-1423.