Intelligent Automation & Soft Computing

Tech Science Press

DOI: 10.32604/iasc.2024.045402

ARTICLE

ABMRF: An Ensemble Model for Author Profiling Based on Stylistic Features Using Roman Urdu

Aiman[1], Muhammad Arshad[1], Bilal Khan[1], Khalil Khan[2], Ali Mustafa Qamar[3,*] and Rehan Ullah Khan[4]

[1]Department of Computer Science, City University of Science and Information Technology, Peshawar, Pakistan

[2]Department of Computer Science, School of Engineering and Digital Sciences, Nazarbayev University, Astana, Kazakhstan

[3]Department of Computer Science, College of Computer, Qassim University, Buraydah, Saudi Arabia

[4]Department of Information Technology, College of Computer, Qassim University, Buraydah, Saudi Arabia

*Corresponding Author: Ali Mustafa Qamar. Email: al.khan@qu.edu.sa

Received: 25 August 2023 Accepted: 27 December 2023 Published: 21 May 2024

ABSTRACT

This study explores the area of Author Profiling (AP) and its importance in several industries, including forensics, security, marketing, and education. A key component of AP is the extraction of useful information from text, with an emphasis on the writers' ages and genders. To improve the accuracy of AP tasks, the study develops an ensemble model dubbed ABMRF that combines AdaBoostM1 (ABM1) and Random Forest (RF). The work uses an extensive technique that involves text message dataset pretreatment, model training, and assessment. To evaluate the effectiveness of several machine learning (ML) algorithms in classifying age and gender, including Composite Hypercube on Random Projection (CHIRP), Decision Trees (J48), Naïve Bayes (NB), K Nearest Neighbor, AdaboostM1, NB-Updatable, RF, and ABMRF, they are compared. The findings demonstrate that ABMRF regularly beats the competition, with a gender classification accuracy of 71.14% and an age classification accuracy of 54.29%, respectively. Additional metrics like precision, recall, F-measure, Matthews Correlation Coefficient (MCC), and accuracy support ABMRF's outstanding performance in age and gender profiling tasks. This study demonstrates the usefulness of ABMRF as an ensemble model for author profiling and highlights its possible uses in marketing, law enforcement, and education. The results emphasize the effectiveness of ensemble approaches in enhancing author profiling task accuracy, particularly when it comes to age and gender identification.

KEYWORDS

Machine learning; author profiling; AdaBoostM1; random forest; ensemble learning; text classification

1 Introduction

The author's profile (AP) can only be formatted as an article, like a bunch of text, and must differentiate between age, gender, the local language, profession, education, and comparative personality traits [1]. AP and author identification are two tasks in the automated extraction of author-related information from textual material. Moreover, in case of AP, demographics are essential; gender and age are two examples of obtaining data. In the case of author identification, the objective is to find out who wrote what, e.g., predict the author of a text from a pool of candidates [2]. The research in

the AP domain shows that the language features of Facebook, status updates, tweets, messages, and blog posts allow us to accurately evaluate the age and gender of authors [3]. AP is an important task in various fields, including information, such as forensics, security, medicine, and marketing [4,5]. It is important to know the author's profile of the harassment message. Additionally, from a marketing point of view, organizations will get to know each other by analyzing online websites and items. What kind of people comment on their items, and will they promote their efforts toward a specific gender or age limit [6]? AP is also used in the educational domain, for instance, researchers can find out the level of knowledge of exceptionally talented students by analyzing their writings. In the case of literary and historical studies, AP can be applied to confirm the characteristics of the author of a text [7]. The goal of AP is to learn as much as possible about a person through analysis of their posts [8]. It is usual to provide false names, gender, age, and location on social media to conceal one's true identity.

AP is a research area that many researchers have focused on to obtain details by analyzing authors' written texts. Authorship analysis is accomplished through authorship identification, plagiarism detection, and AP. Firstly, authorship identification is the process of determining who wrote a particular work. Plagiarism detection, on the other hand, recognizes the author's contribution to the given material. In the present information era, AP is an important technique. It is used in forensic analysis, marketing, and security [9,10]. While communicating with a friend through WhatsApp, sharing on Facebook, posting on Twitter, or composing a blog entry, users invariably create digital traces in the shape of textual information. Studies in the field of author profiling have demonstrated that the linguistic attributes found in Facebook status updates and, blog articles enable precise deductions regarding the age and gender of the authors. To identify internet predators, both social network moderators and law enforcement agencies are working on these issues [11]:

1. Manually analyzing a large number of communications and profiles on a social network is impossible.
2. Internet hackers often use false identities to contact their victims. As a result, a well-designed automated method for identifying and testing is becoming increasingly important.
3. The automatic extraction of information from text identified with the gender, age, and other segment attributes of the author is fundamental in criminology, security, and advertising. For example, someone might seek to understand the language patterns used by the sender of aggressive text messages. Similarly, businesses might aim to learn from reviews left by customers who either favor or criticize their products, utilizing online surveys as sources for analysis [9].

Author profiling research is motivated by its wide-ranging forensics, security, marketing, and personalized content delivery applications. By analyzing linguistic features, researchers can infer valuable insights about an author's age, gender, occupation, and education. This information aids law enforcement agencies in identifying anonymous authors involved in criminal activities, allows businesses to tailor their messages to specific target audiences, and enables the customization of the content based on individual preferences. Moreover, AP contributes to psychological and sociolinguistic understanding and advances in computational linguistics, leading to improved text classification and natural language processing algorithms. The main contributions of this study revolve around enhancing the accuracy of age and gender prediction. Firstly, it achieves improved accuracy in predicting age and gender. Additionally, the study introduces an ensemble model approach for predicting age and gender, offering a novel and robust method.

The rest of the paper is organized as follows: Section 2 presents the brief literature review, Section 3 discusses the proposed model and experimental setup, and Section 4 illustrates the results and discussion. Finally, Section 5 concludes the study.

2 Related Work

Sittar et al. [12] used the FIRE'18-MAP on the SMS dataset for the AP. They used a multilingual (English and Roman Urdu) SMS-based document. They carried out different experiments, i.e., using all 29 stylistic features, all 14 language-independent stylistic features, and individual language-independent stylistic features. They concluded that RF achieved the best accuracy of 73.71% for the gender, while using all 14 language-independent features together and an accuracy of 58.57% for the age group using all 29 features together. They obtained 55% and 37% accuracy on the testing data for gender and age, respectively. The authors concluded that the best results were obtained by using RF, with an accuracy of 73.71% for the gender age (accuracy = 58.57%) by using Logistic Regression (LR). The overall result was compared with the baseline technique. Ouni et al. [13] focused on the following techniques: K-nearest neighbor (KNN), Support Vector Machine (SVM), LR, Radial Basis SVM function (RBFSVM), SVM Linear (SVML), Convolutional Neural Network (CNN), and the proposed author profiling approach based on statistical features. Seventeen stylometric features were used to train the model. The best accuracy for both detections was 92.45% in the English dataset and 90.36% for the gender classification. For the Spanish dataset, 89.68% and 88.88% accuracy were obtained for bot detection.

Wiegmann et al. [14] used SVM, LR, Deep Pyramid CNN (DPCNN), Naïve Bayes (NB), Gaussian NB (GNB), NB complement (NBC), Random Forest (RF), Region-Based CNN (RCNNs) for the AP. They used stylistic and word-level features. The models do not work well for predicting unfrequented demographics, i.e., non-binary gender or work that is not single-topic (e.g., manager, professional, and science). Predicting the date of birth is most accurate between 1980 and 2000 when the age range is 20–40, but it does not work well for older people. The authors in [6] proposed a technique for predicting age and gender, where they used multi-lingual (English and Spanish) corpus datasets (PAN-2018) and applied RF to classify age and gender. They used different features, i.e., lexical, grammatical category, close words, suffixes, and signs. Precision, recall, and the F1 score are used as evaluation measures. The results obtained using only the training set indicate a more effective gender classification compared to age. However, the F1 measure did not exceed 55% in either case. Furthermore, when combining the two classifiers, this measure decreases resulting in an F1 value between 40% and 44%. Nemati [15] used a combination of semantic, syntactic, and Natural Language Processing (NLP) as a feature. Then, all these combinations are fed into an ensemble model that classifies age and gender. They employed a supervised random forest ensemble classifier for the AP using the PAN2014 dataset. Only working language indicates readability criteria, function words, and structural features play a vital role in identifying the age and gender of the writer.

For predicting an author's profile, Kovács et al. [16] proposed a technique for Twitter bots that can only categorize human gender as female or male. Both users were in 11 Twitter bots; from their profiles, only a hundred tweets were selected overall, and another hundred tweets were chosen randomly. They focused on the semantic feature category, which is present in the tweets. They joined those semantic features with other stylistic features and Part-Of-Speech (POS) tags. They used various ML methods with an ensemble model and determined Adaboost's F1 score to be 99%. For the English language, the results gained an accuracy of 89.17%. The RF technique was employed to predict the profile of an author. In another article, Sapkota et al. [17] described an approach to work with an author's profile for the PAN 2013 Challenge. This work is based on a linguistic method used in other classification tasks, such as document writing. They considered three features: syntactic, stylistic, and semantic. Each represents a different aspects of the text. They extracted similarity relationships between attribute vectors in test files and center-specific modality clusters for each method.

Grivas et al. [18] presented an explicit feature in the form of a group; each group is then put together with appropriate pre-processing steps for each group. The metrics used were structural, trigrams, counts of Twitter's most essential characteristics, and stylometric grouping. The authors clarified that age and gender prediction are classification jobs and character prediction is a regression problem employing SVM and Support Vasomotor Rhinitis (SVMR), respectively. Ashraf et al. [19] focused on age and gender prediction and carried out the experiments using deep learning (DL) methods, i.e., Long Short-Term Memory (LSTM), Gated Recurrent Units (GRU), Bi-LSTM, and CNN. BAT-AP-19, RUEN-AP-17, and SMS-AP-18 corpus datasets were used for training and testing. This research focused on POS features. The best accuracy was achieved when the Bi-LSTM classifier was used. The best scores were achieved as follows: accuracy = 0.882, F1 score = 0.839, accuracy = 0.735, and F1 score = 0.739 for age and gender, respectively, using LSTM. To predict the age, gender, education, language, country, and emotions of AP, Estival et al. [20] used Sequential Minimal Optimization (SMO), RF, bagging, and instance-based learning with parameter k (KNN) and Lib-SVM. The English Email dataset is used in this study to extract character (case word length), lexical (function word correlate), structural (document category HTML), and POS features. The best result was achieved using RF for the language classification, which is 84.22%. In another research [21], the authors proposed a technique for age, gender, and country prediction. They used a linear classifier, SVM, and Multi-Layer Perceptron (MLP) classifier as an ML technique. The researchers used Arabic-shared tasks. There are two tasks, AP in Arabic tweets and deception detection in Arabic texts.

The authors in [3] investigated language differences in instant messages to infer both age and gender, building upon and expanding earlier studies conducted on social media content. Analyzing more than 0.3 Million WhatsApp messages from 226 volunteers, the study employs ML algorithms to predict age and gender with significant accuracy above baseline levels. The results demonstrated the potential for inferring individual characteristics from the instant messaging data, highlighting implications for the psycholinguistic theory, author profiling applications, and concerns over privacy rights in the context of growing private messaging usage and weaker user data protection. The researchers in [22] introduced a novel end-to-end age and gender recognition system for speech signals using CNN with a multi-attention module (MAM). The MAM effectively extracts spatial and temporal salient features from the input data by incorporating separate time and frequency attention mechanisms. Combining these features led to a strong performance for classifying both gender and age. The proposed system achieved significant accuracy scores in gender, age, and combined age-gender classification tasks when tested on the common voice and locally developed Korean speech recognition datasets. The results showed the model's superiority in recognizing age and gender using speech signals.

Suman et al. [23] addressed the challenge of automatically predicting the gender of authors on Twitter using multimodal data. The authors proposed an efficient neural framework that combines text and image information from tweets for gender classification. They utilize BERT_base for the text representation and EfficientNet for image feature extraction, employing a fusion strategy to combine the modalities. The model achieves high accuracies of 82.05% for images, 86.22% for text, and 89.53% for the multimodal setting, surpassing previous state-of-the-art approaches. The study also provided insights into the words that contribute to gender classification.

3 Experimental Setup and Methodology

This section discusses the overall research methodology applied to this study. This study aims to propose an ensemble model for the author's age and gender prediction. All of the authors' text

messages are saved as .txt files by the system. The architecture of the proposed system consists of six parts, as shown in Fig. 1. When the text documents are input into the system, preprocessing steps are applied to each document. After that, four different strategies were compared to predict the author's age and gender by analyzing the author's writing behavior using eight learning models, i.e., RF, AdaboostM1, CHIRP, J48, NB, NB-updatable, and ABMRF, for comparative analysis based on strategy.

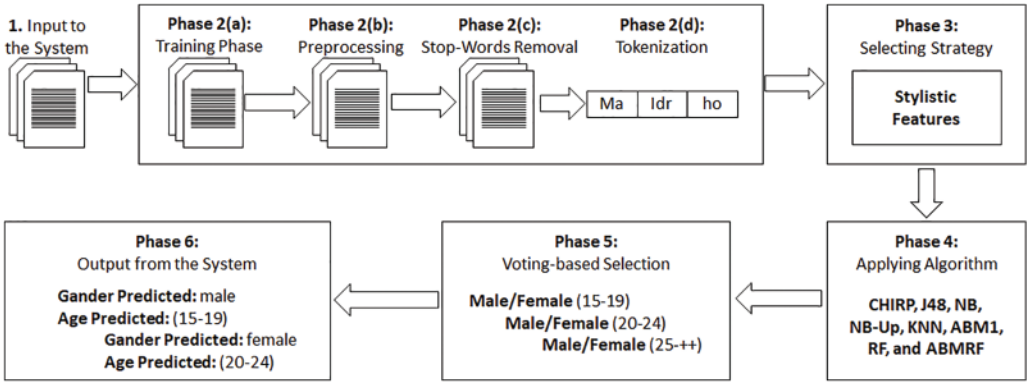

Figure 1: Research methodology

In actual application, most cases are multi-class problems. Boweny and Leesham did not suggest RF in multi-class AdaBoostM1 algorithms, e.g., AdaBoostM1 can do RF in multi-class cases that can fulfill the application of AdaBoostM1, but the correct rate must be better than 50% when both are correctly used [24]. There are two main reasons to choose an ensemble model over a single one, both related. An ensemble model can produce more accurate predictions and perform better than any other contributing model. It can also lower the spread or dispersion of predictions and model performance. As a result, it focused on "boosting", which is an ensemble technique for attempting to construct a strong classifier from a set of weak classifiers. It may also increase a model's overall robustness or dependability.

In the fifth phase, select the top file and predict the class for the testing file. The class assigned to the test files based on the highest number of votes will be against that class. The output from the system is the predicted class of gender and age for the test files. The pseudocode for ABMRF is shown next.

Algorithm 1: Ensembled ABMRF

Initialize $D_1\ (i)$
For $i = 1$ to m such that $D_1\ (i) = 1/m$
  For $t = 1$ to T:
    h_i = None
      For $K = 1$ to K:
        ${}^{\theta}k$ = GenerateVector ()
        $h(x,{}^{\theta}k)$ using any Decision Tree Algorithm
        tree = ConstructDecisionTree $(T,{}^{\theta}k)$
        if h_i is None:
          h_i = tree
        else:
          h_i = CombineHypotheses (h_i, tree)

(Continued)

Algorithm 1 (continued)

Return h_i

Get back hypothesis $h_t: X \rightarrow Y$ error:

$$\epsilon_t = \sum_{i h_t (X_t) \neq y_i} D_t (i)$$

If $\epsilon_t > 1/2$, then set $T = t\text{-}1$ and abort loop

- Set $\beta_t = \dfrac{1}{1 - \epsilon_t}$
- Update D_t :

$$D_i + 1 (i) = \frac{D_{ti}}{Z_t} * \begin{cases} \beta_t \, if \, h_t = (x_i) = y_i \\ 1 \, if \, h_t = (x_i) = y_i \end{cases}$$

Output: $H (x) = \arg\max_{y \epsilon Y} + \sum_{t: h_t(x)=y}^{t} \left(\log \dfrac{1}{\beta} \right)$

3.1 Proposed Methodology

The proposed model is based on AdaboostM1 (ABM1) and Random Forest (RF) to achieve better analysis than other employed models. ABM1 is a popular ML model used for classification tasks. It is an ensemble learning method that combines multiple weak classifiers to create a strong classifier. It operates by iteratively training a sequence of weak classifiers, assigning higher weights to misclassified instances in each iteration. The weak classifiers are typically simple classifiers, such as decision stumps. During training, ABM1 adjusts the weights of the training instances based on their classification accuracy. The subsequent weak classifiers focus more on the previously misclassified instances, allowing them to correct the mistakes made by previous weak classifiers. The final prediction is determined by combining the predictions of all weak classifiers, weighted by their accuracy. It can be calculated as:

 # Step 1: Initialize weights of training instances

InitializeWeights (W, N) # where N is the number of training instances

for i in range (N):

$W[i] = 1/N$

 # Step 2: AdaBoost Iterations

for t in range (T): # where T is the number of iterations

 # Step 2a: Train a weak classifier

h_t = TrainWeakClassifier (X, Y, W) # X is the training data, Y represents the labels

 # Step 2b: Compute the weighted error

$epsilon_t$ = ComputeWeightedError (h_t, X, Y, W)

 # Step 2c: Calculate the weight of the weak classifier

$alpha_t = 0.5 * ln \; ((1 - epsilon_t)/epsilon_t)$

 # Step 2d: Update the weights of training instances

UpdateWeights $(W, h_t, X, Y, alpha_t)$

Normalize the weights

W = NormalizeWeights (W)

 # Step 3: Combine weak classifiers into a strong classifier

def StrongClassifier (X, T, classifiers, alphas):

$H = 0$

for t in range(T):

H += $alphas[t] * classifiers[t](X)$

return $sign(H)$

In the aforementioned equations, x_i represents the feature vector of instance i, and y_i represents its true label. $h_t(x_i)$ represents the prediction of weak classifier h_t , for instance, i. The sign function returns $+1$ for positive predictions and -1 for negative predictions. The proposed ensemble model ABMRF works as follows:

 # Step 1: Initialize the weights of training instances

InitializeWeights (W, N)

for i in range (N):

$W[i] = 1/N$

 # Step 2: AdaBoost Iterations (ABM1)

for t in range (T): # where T is the number of iterations

 # Step 2a: Train a weak classifier

h_t = TrainWeakClassifier (X, Y, W)

 # Step 2b: Compute the weighted error

$epsilon_t$ = ComputeWeightedError (h_t, X, Y, W)

 # Step 2c: Calculate the weight of the weak classifier

$alpha_t = 0.5 * ln((1 - epsilon_t)/epsilon_t)$

 # Step 2d: Update the weights of the training instances based on misclassifications

UpdateWeights(W, h_t, X, Y, $alpha_t$)

 # Step 2e: Normalize the weights

W = NormalizeWeights (W)

 # Step 3: Combine ABM1 with RF

for t in range (T):

 # Step 3a: Train a Random Forest classifier with the current weights

RF_t = TrainRandomForest (X, Y, W)

 # Step 3b: Compute the weighted error

$psilon_t$ = ComputeWeightedError (RF_t, X, Y, W)

 # Step 3c: Calculate the weight of the RF classifier

$alpha_t = 0.5 * ln((1 - epsilon_t)/epsilon_t)$

 # Step 3d: Update the weights of the training instances based on misclassifications

```
    UpdateWeights (W, RF_t, X, Y, alpha_t)
    # Step 3e: Normalize the weights
    W = NormalizeWeights (W)
# Step 4: Combine the weak classifiers (RFs) into a strong classifier
def StrongClassifier (X, T, classifiers, alphas):
    H = 0
    for t in range (T):
        H += alphas[t] * classifiers[t] (X)
    return sign (H)
```

Fig. 2 presents the flow diagram of the proposed model.

First, input train data is provided, and the data is initialized with labels. Later, weak learners are trained using distribution D_t. Then, select the number of trees (K) to be constructed. Construct the Tree Algorithm $h\ (x,\ O)/K$ employing any decision tree. Each tree contributes a vote for the most prevalent class at X. Prediction of the class at X involves choosing the one with the maximum votes. The process results in returning the hypothesis h_i. Lastly, retrieve the hypothesis h: X–Y.

The details of the features used in the proposed model based on ABM1 and RF are as follows:

1. Bag Size Percent: ABM1 employs a bag size percent of 100, while RF utilizes 0.
2. Batch Size: ABM1 and RF employ a batch size of 100.
3. Number Decimal Places: ABM1 and RF utilize two decimal places.
4. Number Execution Slots: ABM1 uses 0 execution slots, whereas RF uses 1.
5. Number Iterations: ABM1 undergoes 10 iterations, whereas RF undergoes 100.
6. Seeds: Both ABM1 and RF use a single seed each.
7. Weight Threshold: ABM1 employs a weight threshold of 100, while RF employs 0.

3.2 Stylistic-Based Features

Words or sentences are arranged to provide understanding, including narrative perspectives, stanza structure, and positioning as stylistic features. Table 1 shows a total of fourteen stylistic features. The code for feature extraction can be found at: https://github.com/aiman-syed/Features-Extraction.

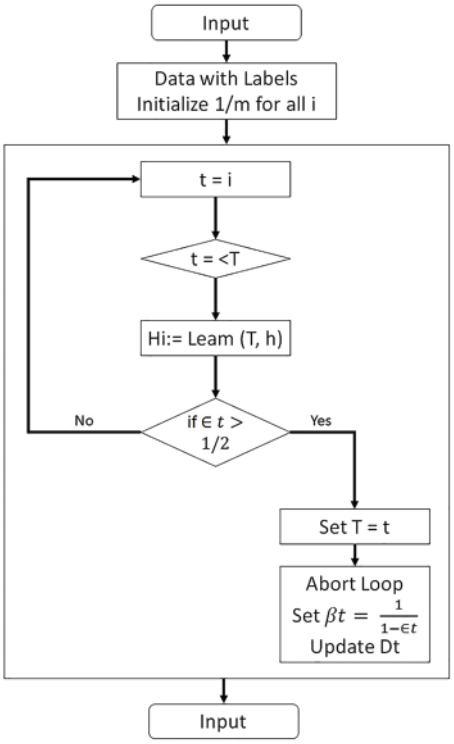

Figure 2: Proposed ensemble ABMRF

Table 1: Stylistic features with descriptions

Features	Description
Average word length	First, it counts the words used in a text document and then finds its average.
Average sentence length	First, it counts the total number of sentences used in a text document and then finds its average.
Percentage of words with six and more letters	Percentage of only words having six or more letters.
Percentage of words with two and three letters	Words containing two and three letters.
Percentage of question sentences	Percentage of question sentences in a text document.
Percentage of semicolons	Percentage of semicolon sentences.
Percentage of punctuations	Percentage of punctuations used.
Percentage of comma	Percentage of comma sentences used.
Percentage of short sentences	It counts only those sentences with a word length of less than eight and then finds the percentage of only those sentences.
Percentage of long sentences	It counts only those sentences with a word length greater than 15 and then finds the percentage of those sentences.
Percentage of capitals	Percentage of capital letters employed in a text document.

(Continued)

Table 1 (continued)

Features	Description
Percentage of colons	Percentage of colons used.
Percentage of digits	Percentage of digits used.
Percentage of full stop	Percentage of full-stop sentences employed.

3.3 Experimental Setup

The application runs on Google Colaboratory (Colab), which offers free CPU cloud services powered by TensorFlow. Google Colab provides a 33 GB hard drive, 12 GB of RAM, and a 2-core Xeon 2.2 GHz processor. The system is tested using Python 3.x (3.6). Stylometric features are used to generate output, which is input to machine learning (ML) algorithms. In our experimental setup, our model utilizes 10-fold cross-validation to assess its performance. Experiments are conducted using eight different ML algorithms: NB, J48, KNN, CHIRP, AdaBoostM1, NB Updatable, RF, and ABMRF.

3.3.1 Dataset Description

This study focuses on the FIRE'18 MAP on SMS and the Roman Urdu dataset. The FIRE'18 MAP SMS dataset comprises both testing and training instances. The training set includes 350 instances and one truth file that specifies gender as either male or female, while age is further divided into the following groups: 15–19, 20–24, and 25–XX. In the training set, there are 210 males, 140 females, and 18 unspecified files. Age groups are distributed as follows: 15 to 19 (108 records), 20 to 24 (176 records), and 25 to XX (XX implies an open-ended age range) with 66 records. The testing set consists of 150 instances, which will be used to evaluate the proposed research model.

3.3.2 Pre-Processing Steps

Pre-processing typically consists of three phases, and traditionally, text pre-processing has been a crucial step in NLP. It simplifies language to enable machine learning algorithms to operate more effectively. The dataset underwent the following three pre-processing steps.

a. Removal of Whitespaces

Almost all textual data in the world contains white space, which, when removed, enhances readability and ease of understanding [19]. The *strip( )* function in Python is used to eliminate leading and trailing spaces from a text line.

b. Stop Words Removal

Stop words have no significant semantic relation to the context in which they exist [20]. During this phase, stop words are removed, as shown in Fig. 3. In Part A, the text includes 'Mai' and 'hai,' both considered stop words. These stop words are removed in Part B during the text preprocessing.

c. Tokenization

It is the method by which text is broken down into smaller pieces known as tokens. Tokens typically include words, numbers, and punctuation marks. The sentences are tokenized using Python's split() function in the preprocessing phase.

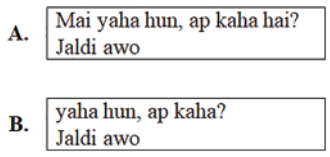

Figure 3: Stop words removal

3.3.3 Training and Evaluation

Model training and testing are the core phases of any ML-based analysis. To this end, the study employs 10-fold cross-validation [25]. The performance of the proposed and other models is evaluated using standard assessment measures, including accuracy, precision, recall, F-measure, and Matthews Correlation Coefficient (MCC) [26–28]. These measures can be calculated as shown in Eqs. (1)–(5).

$$\text{Accuracy} = \frac{\text{TP} + \text{TN}}{\text{TP} + \text{TN} + \text{FP} + \text{FN}} \tag{1}$$

$$\text{Precision} = \frac{\text{TP}}{\text{TP} + \text{FP}} \tag{2}$$

$$\text{Recall} = \frac{\text{TP}}{\text{TP} + \text{FN}} \tag{3}$$

$$\text{F} - \text{measure} = \frac{2 * \text{Precision} * \text{Recall}}{\text{Precision} + \text{Recall}} \tag{4}$$

$$\text{MCC} = \frac{\text{TP} * \text{TN} - \text{FP} * \text{FN}}{\sqrt{(\text{TP} + \text{FP})\,(\text{TP} + \text{FN})\,(\text{TN} + \text{FP})\,(\text{TN} + \text{FN})}} \tag{5}$$

4 Experimental Results and Analysis

The following declared terminologies are used here. The term "Classifier" indicates the ML algorithm applied to produce the numeric scores, such as NB, NB Updatable, J48, KNN, RF, CHIRP, AdaBoostM1, and ABMRF. Fourteen stylistic features, as depicted in Table 1, are used in this study. All evaluations, including precision, recall, F-measure accuracy, and MCC, are derived from the confusion matrix (CM). The CM for gender using stylistic features is shown in Table 2.

Table 2: Confusion matrix for *Gender* using stylistic features

Technique	Class	a = Male	b = Female
J48	a = Male	170	40
	b = Female	68	72
RF	a = Male	193	17
	b = Female	87	53
NB	a = Male	167	43
	b = Female	105	35
IBK	a = Male	154	56
	b = Female	64	76

(Continued)

Table 2 (continued)

Technique	Class	a = Male	b = Female
CHIRP	a = Male	1	208
	b = Female	3	137
NB updatable	a = Male	187	23
	b = Female	113	27
AdaBoostM1	a = Male	172	38
	b = Female	75	65
ABMRF	a = Male	192	18
	b = Female	83	57

Table 3 shows the outcomes via precision-recall and F-measure for age and gender. This also indicates the better performance of age using ABMRF, with outcomes of 0.720, 0.543, 0.420, and 0.723, 0.711, 0.687 for precision, recall, and F-measure for age and gender, respectively. Here, it demonstrated the average values for each attribute. Conversely, NB produces the poorest results in precision, recall, and F-measure, which are 0.309, 0.395, and 0.232, respectively. Some tables, such as Table 3, contain a (?) sign. In these cases, the question mark sign is used instead of the message "DIV/0!" due to the value "0" in it. In the confusion matrix, according to different equations, division cannot be performed when certain values need to be divided and that value becomes "0".

Table 3: Precision, Recall, and F-measure for *Age* and *Gender* using stylistics features

Algorithm	Age			Gender		
	Precision	Recall	F-measure	Precision	Recall	F-measure
J48	0.563	0.537	0.41	0.716	0.703	0.675
RF	0.456	0.48	0.46	0.686	0.691	0.684
NB	0.483	0.491	0.483	0.309	0.395	0.232
IBK	?	0.529	?	0.67	0.677	0.666
CHIRP	0.409	0.354	0.344	0.59	0.611	0.554
NB updatable	0.562	0.516	0.368	0.654	0.657	0.655
AdaBoostM1	0.409	0.354	0.344	0.590	0.611	0.554
ABMRF	**0.720**	**0.543**	0.420	**0.723**	**0.711**	**0.687**

Moreover, Table 4 shows the outcomes of each technique using MCC and accuracy. These analyses demonstrate the superior performance of ABMRF, getting an accuracy of 54.29% for age and 71.14% for gender. However, using stylistic features, CHIRP exhibits the weakest performance.

proposed model is better than Comparing ABMRF with other techniques, the best result was achieved by ABMRF, i.e., 54.29% for age. On the other hand, the worst performance of NB and NB Updatable is noted with an accuracy of 35.4%, while ABMRF achieved the best result of 71.14%. In contrast, the worst performance of CHIRP is pointed out, with an accuracy of 39.54% for gender. The

accuracy of the proposed model is better than that of the other models used in this research, as shown in Table 4, as well as in the work of Sittar et al. [13].

Table 4: MCC and accuracy for stylistic features

Algorithm	MCC		Accuracy	
	Age	Gender	Age	Gender
J48	0.168	0.365	0.480	0.691
RF	0.108	0.34	0.537	0.703
NB	0.097	−0.077	0.354	0.611
IBK	?	0.305	0.491	0.657
CHIRP	0.189	0.117	0.516	0.395
NB updatable	0.146	0.279	0.354	0.611
AdaBoostM1	−0.003	0.117	0.529	0.677
ABMRF	**0.201**	**0.384**	**0.543**	**0.711**

Fig. 4 illustrates the percentage difference (PD) in Age between the superior technique and the other method employed in the study. The PD is calculated as shown in Eq. (6).

$$PD = \left(\frac{n1 - n2}{\frac{n1 + n2}{2}} \right) * 100 \tag{6}$$

where n_1 represents the value of ABMRF and n_2 stands for the value of other techniques. For stylistic features, the illustration shows that the minimal difference between ABMRF and RF is 1.07%, and the highest difference between ABMRF and NB is 42.03%.

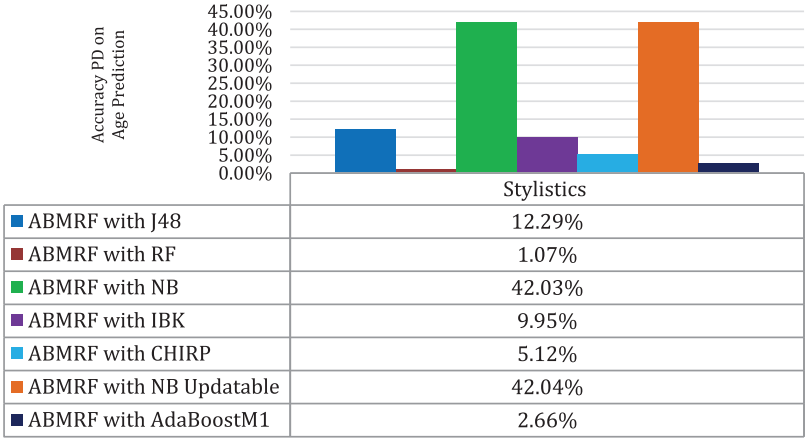

Figure 4: Percentage difference (PD) for age

Fig. 5 shows the PD for gender using stylistic features. The illustration shows that the minimal difference between ABMRF and RF is 1.21%, while the most significant difference between ABMRF and CHIRP is 57.10%.

There are two primary reasons to prefer an ensemble model over a single one, and they are interconnected. An ensemble model can outperform any contributing model in terms of performance and accuracy. It may also stop the prevalence or dispersion of predictions and model performance. As a result, focused on boosting, an ensemble strategy for attempting to build a strong classifier from a set of weak classifiers. It may also improve the overall robustness or reliability of a model. With AdaBoostM1, create an ensemble with RF. In an ML project, these are significant considerations, and one may sometimes choose one or both qualities from a model. The following are some reasons for selecting the ABMRF: It gives variable weight, which helps identify the variable with a beneficial influence. ML models are frequently overfitted, but RF classifiers are not. There are different amounts of text in each file in this situation. Furthermore, when a class is rarer than other classes in the data, as in this case, it may automatically balance datasets. Ensemble classifiers also outperform nonlinear classifiers on a wide range of tasks.

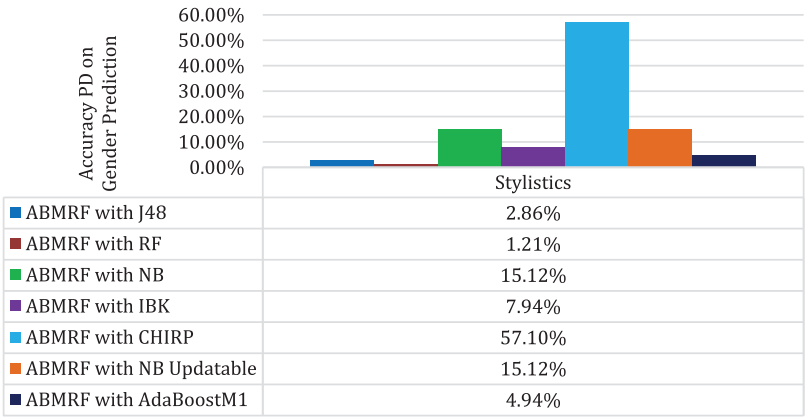

Figure 5: PD for Gender

As mentioned in the analysis, ABMRF outperforms other classifiers to increase accuracy. This study focuses on eight diverse ML techniques for AP detection. The methods are evaluated using multiple assessment measures on the Fire'18 MAP SMS dataset. Compared with CHIRP, J48, RF, NB, IBK, AdaboostM1, NB Updatable, and ABMRF techniques, we found that the Ensemble ABMRF is the most optimal for age and gender classification in AP. Our experiments depict the better performance of ABMRF for age and gender, which achieved an accuracy of 54.29% and 71.14%, respectively. The worst performance of CHIRP is noted, with an accuracy of 35.43% and 39.54% for age and gender, respectively.

5 Conclusion

This study has examined AP with an emphasis on age and gender identification, demonstrating the importance of this discipline in a variety of industries, including forensics, security, marketing, and education. The study developed an innovative ensemble model, ABMRF, which combines ABM1 and RF to improve the precision and efficiency of AP tasks. This study compared ABMRF with several different ML algorithms using a careful technique that included data pretreatment, model training, and an in-depth assessment of a dataset made up of text messages. With an accuracy percentage of 54.29% for age classification and 71.14% for gender classification, the data unmistakably showed that ABMRF consistently outperformed its competitors. ABMRF performed exceptionally well in both age and gender profiling tasks, as evidenced by measures like precision, recall, F-measure,

MCC, and accuracy. These results highlight the critical function of ensemble approaches in improving the precision and dependability of author profiling, especially in the complex areas of age and gender identification. The effectiveness of ABMRF has wide-ranging effects, helping businesses with targeted marketing, educational institutions assess students' knowledge levels, and law enforcement organizations follow cyber criminals. This study sheds light on the ever-expanding horizons of AP applications by illuminating the transformational potential of ABMRF in furthering the disciplines of computational linguistics, text categorization, and natural language processing. Author profiling is developing as a vital tool for identifying and comprehending people via their written expressions in a world becoming increasingly controlled by digital interactions and social media. ABMRF is at the forefront of this technical innovation.

However, it is important to acknowledge certain limitations of the proposed model. Firstly, the model interpretability is reduced, and incorrect selections may lead to decreased predictive accuracy compared to individual models. Additionally, feature tuning proves challenging, when integrating RF and AdaboostM1 models in an ensemble. Furthermore, the model requires considerable resources in terms of space and time, resulting in higher computational costs.

Future research directions in AP may involve broadening the focus to incorporate more demographic characteristics, incorporating deep learning strategies, addressing privacy issues, investigating cross-lingual profiling, and adjusting to new digital communication platforms, allowing a more thorough and flexible approach to understanding people through their online content. Furthermore, further investigations should involve larger datasets to expand the scope of analysis. Moreover, using deep learning techniques can lead to better results. By doing so, a more comprehensive understanding of author profiling in Roman Urdu can be achieved, ultimately improving accuracy and effectiveness.

Acknowledgement: The researchers would like to thank the Deanship of Scientific Research, Qassim University for funding the publication of this project.

Funding Statement: The authors received no specific funding for this study.

Author Contributions: The authors confirm their contribution to the paper as follows: study conception and design: Aiman, Bilal Khan, Muhammad Arshad; data collection: Aiman, Muhammad Arshad, Rehan Ullah Khan; analysis and interpretation of results: Aiman, Bilal Khan, Khalil Khan; draft manuscript preparation: Ali Mustafa Qamar, Khalil Khan. All authors reviewed the results and approved the final version of the manuscript.

Availability of Data and Materials: The data used in this study is available at: https://lahore.comsats.edu.pk/cs/MAPonSMS/de.html.

Conflicts of Interest: The authors declare that they have no conflicts of interest to report regarding the present study.

References

[1] I. Ameer, G. Sidorov, and R. M. A. Nawab, "Author profiling for age and gender using combinations of features of various types," *J. Intell. Fuzzy Syst.*, vol. 36, no. 5, pp. 4833–4843, 2019.

[2] J. Soler-Company and L. Wanner, "On the relevance of syntactic and discourse features for author profiling and identification," in *Proc. 15th Conf. Eur. Chapter Assoc. Comput. Linguist.*, Valencia, Spain, 2017, pp. 681–687.

[3] T. K. Koch, P. Romero, and C. Stachl, "Age and gender in language, emoji, and emoticon usage in instant messages," *Comput. Hum. Behav.*, vol. 126, pp. 106990, 2022. doi: 10.1016/j.chb.2021.106990.
[4] F. C. Hsieh, R. F. S. Dias, and I. Paraboni, "Author profiling from Facebook corpora," in *Proc. Int. Conf. Lang. Resour. Eval.*, Miyazaki, Japan, 2018, pp. 2566–2570.
[5] G. Farnadi, J. Tang, M. de Cock, and M. F. Moens, "User profiling through deep multimodal fusion experimental results: User profiling," in *Proc. Int. Conf. Web Search Data Mining*, Marina Del Rey, CA, USA, 2018, pp. 171–179.
[6] J. Silva *et al.*, "A method for detecting the profile of an author," *Procedia Comput. Sci.*, vol. 170, pp. 959–964, 2020.
[7] D. Radha and P. C. Sekhar, "Author profiling using stylistic and N-gram features," *Int. J. Eng. Adv. Technol.*, vol. 9, no. 1, pp. 3044–3049, 2019.
[8] E. R. D. Weren *et al.*, "Examining multiple features for author profiling," *J. Inf. Data Manag.*, vol. 5, no. 3, pp. 266, 2014.
[9] A. Sboev, T. Litvinova, D. Gudovskikh, R. Rybka, and I. Moloshnikov, "Machine learning models of text categorization by author gender using topic-independent features," *Procedia Comput. Sci.*, vol. 101, pp. 135–142, 2016.
[10] T. R. Reddy, B. V. Vardhan, and P. V. Reddy, "N-gram approach for gender prediction," in *Proc. IEEE 7th Int. Adv. Comput. Conf. (IACC)*, Hyderabad, India, 2017, pp. 860–865.
[11] C. Peersman, W. Daelemans, and L. V. Vaerenbergh, "Predicting age and gender in online social networks," in *Proc. 3rd Int. Workshop Search Mining User-Gen. Contents*, Glasgow, Scotland, UK, 2011, pp. 37–44.
[12] A. Sittar and I. Ameer, "Multi-lingual author profiling using stylistic features," in *Proc. Working Notes FIRE, 2018-Forum Inf. Retrieval Eval.*, Gandhinagar, India, 2018, pp. 240–246.
[13] S. Ouni, F. Fkih, and M. N. Omri, "Toward a new approach to author profiling based on the extraction of statistical features," *Soc. Netw. Anal. Mining*, vol. 11, no. 1, pp. 1–16, 2021. doi: 10.1007/s13278-021-00768-6.
[14] M. Wiegmann, B. Stein, and M. Potthast, "Overview of the celebrity profiling task at PAN 2019," in *Proc. Working Notes Conf. Labs Eval. Forum*, Lugano, Switzerland, vol. 2380, 2019.
[15] A. Nemati, "Gender and age prediction multilingual author profiles based on comments," in *Proc. Working Notes FIRE*, Gandhinagar, India, 2018.
[16] G. Kovács, V. Balogh, P. Mehta, K. Shridhar, P. Alonso and M. Liwicki, "Author profiling using semantic and syntactic features," in *Notebook PAN CLEF 2019, Proc. Working Notes Conf. Labs Eval. Forum*, Lugano, Switzerland, vol. 2380, 2019.
[17] U. Sapkota, T. Solorio, M. Montes-y-Gómez, and G. Ramírez-de-la-Rosa, "Author profiling for English and Spanish text," in *Notebook PAN CLEF 2013, Proc. Working Notes Conf. Labs Eval. Forum*, Valencia, Spain, vol. 1179, 2013.
[18] A. Grivas, A. Krithara, and G. Giannakopoulos, "Author profiling using stylometric and structural feature groupings notebook for PAN at CLEF 2015," in *Proc. Working Notes Conf. Labs Eval. Forum*, Toulouse, France, 2015.
[19] M. A. Ashraf, R. M. A. Nawab, and F. Nie, "Author profiling on bi-lingual tweets," *J. Intell. Fuzzy Syst.*, vol. 39, no. 2, pp. 2379–2389, 2020. doi: 10.3233/JIFS-179898.
[20] D. Estival, T. Gaustad, S. B. Pham, W. Radford, and B. Hutchinson, "Author profiling for English emails," in *Proc. Int. Conf. Pacific Assoc. Comput. Linguist.*, Melbourne, Australia, 2007, pp. 263–272.
[21] H. A. Nayel, "NAYEL @ APDA: Machine learning approach for author profiling and deception detection in Arabic texts," in *Proc. Working Notes FIRE 2019–Forum Inf. Retrieval Eval.*, Kolkata, India, 2019, pp. 92–99.
[22] A. Tursunov, J. Y. Choeh Mustaqeem, and S. Kwon, "Age and gender recognition using a convolutional neural network with a specially designed multi-attention module through speech spectrograms," *Sens.*, vol. 21, no. 17, pp. 5892, 2021. doi: 10.3390/s21175892.
[23] C. Suman, A. Naman, S. Saha, and P. Bhattacharyya, "A multimodal author profiling system for tweets," *IEEE Trans. Comput. Soc. Syst.*, vol. 8, no. 6, pp. 1407–1416, 2021. doi: 10.1109/TCSS.2021.3082942.

[24] Z. Zhang and X. Xie, "Research on AdaBoost.M1 with random forest," in *Proc. 2nd Int. Conf. Comput. Eng. Technol. (ICCET)*, Chengdu, China, 2010, pp. V1-647–V1-652.
[25] B. Khan, R. Naseem, F. Muhammad, G. Abbas, and S. Kim, "An empirical evaluation of machine learning techniques for chronic kidney disease prophecy," *IEEE Access*, vol. 8, pp. 55012–55022, 2020. doi: 10.1109/ACCESS.2020.2981689.
[26] C. D. Morales-Molina, D. Santamaria-Guerrero, G. Sanchez-Perez, H. Perez-Meana, and A. Hernandez-Suarez, "Methodology for malware classification using a random forest classifier," in *Proc. Int. Autumn Meeting Power, Electron. Comput. (ROPEC)*, Ixtapa, Mexico, 2019, pp. 1–6.
[27] J. Zhang *et al.*, "A survey on bug-report analysis," *Sci. China Inf. Sci.*, vol. 58, pp. 1–24, 2015.
[28] A. Alajmi, E. M. Saad, and R. R. Darwish, "Toward an Arabic stop-words list generation," *Int. J. Comput. Appl.*, vol. 46, no. 8, pp. 8–13, 2018.