International Journal of Computer Science & Communication (ISSN: 0973-7391)
Volume 17 • Issue 2 pp. 222-233 March 2026-Sept 2026 www.csjournals.com

Predicting Gender by using First Name with Machine Learning

Nisha, Dr. Sneha, Dr. Anupam Bhatia

[1]MCA Student, [3]Asst. Prof., [2]Associate Professor
Deptt. of Comp. Sc. and Application, Chaudhary Ranbir Singh University, Jind, Haryana, India

ABSTRACT

Gender prediction from personal names is a well-studied problem in Natural Language Processing (NLP) applications in demographic analytics, personalization, author profiling, and bias auditing of automated systems. This paper presents an empirical comparative study of five classical machine learning algorithms random forest, decision tree, multinomial naïve bayes, linear support vector machine, logistic regression used in classification of gender on the "Indian and English Names" dataset. After cleaning, normalization, and de-duplication, the working dataset consists of 108,910 unique name gender pairs (66,535 female, 42,375 male). Names are represented using character level TF-IDF vectorization (analyzer=char_wb, n-grams of size 2–5, max_features=8,000, sublinear term-frequency scaling), and class imbalance in the training partition is corrected using SMOTE rather than class weighting. All five models are estimated using f1-score, precision, recall, accuracy on a held out, unaltered 20% test split after trained on the SMOTE balanced training data. The best overall test performance is attained by linear SVM. (88.42% accuracy, 90.31% F1-score), closely followed by Logistic Regression (87.39% accuracy), while Random Forest, despite the strongest precision, lags in recall and overall accuracy (83.02%). The final deployed model is a LinearSVC classifier trained on the SMOTE balanced TF-IDF representation, used for single-name, real-time gender prediction. Our results confirm that character level n gram features combined with synthetic oversampling provide a computationally inexpensive and effective approach to gender prediction on multicultural name datasets.

Keywords: Machine Learning, Natural Language Processing, tf-idf vectorizer, random forest, decision tree, naïve bayes, linear support vector machine, logistic regression.

Introduction

It is not an easy task to predict gender by the first name of a person. This is a requirement that appears in numerous applications particularly in Natural Language Processing (NLP),mainly when foreign names are present. To evaluate the model performances we have f1 score, confusion matrix, recall, accuracy, and precision. The findings demonstrate that the gender prediction may be carried out using the feature extraction approach by considering the names as a collection of strings. Certain methods can accurately estimate gender in more than 95% of the time [1].

Statistical analysis of social, economic and actional data concerning gender helps identify areas where gender bias and inequality exists, and encourages greater participation of women. “We can automatically analyze gender, so it can be faster and easier to study differences in fields like technology, media and education.” It also can improve applications like machine translation, name recognition etc. [2].

It is impossible to always decide the gender of an individual correctly according to their name only. Names are often ambiguous, used by both genders, or unfamiliar to people of other cultures, so no human could ever predict an individual's gender with 100% accuracy according to their name. Sometimes names can be used to identify someone, because they tend to follow certain language patterns. These patterns give an idea about the gender of a person, as some names are used for both male or female. However, these suggestions or ideas are not always correct, since many names can be used by any gender or may be different in different cultures [4].

Today a huge amount of data produced by online applications and some other platforms, or many systems need to understand the basic information about people. One such task is finding out whether a person is male or female just by looking at their name. This is useful in many areas like marketing, social media analysis, recommendation systems and improving user experience. Names often give suggestions about gender because of language and cultural patterns. However, this task is not easy to implement. Many names are used for both males and females, and different cultures have different naming styles. Because of this complexity existing traditional methods do not work well and cannot predict highly accurate value for ambiguous dataset. In order to handle this issue more precisely, machine learning and deep learning approaches are utilized.

For instance, machine learning can use data already in existence to determine a person's gender based only on their name. To improve accuracy, it makes use of sophisticated methods like Random Forest and XGBoost in addition to algorithms like Logistic Regression, SVM, K-NN, and Decision Tree. Machine learning comes in three primary varieties. Supervised learning makes predictions based on labeled data. Unsupervised learning (identifies patterns in unlabeled data). Reinforcement learning is the process of learning by making mistakes and applying incentives and penalties.

Related Work

Author & Year	Dataset Used	Technique &Models Used	Key Features	Results / Accuracy	Limitations
Rego et al. (2021)	Dataset of first names	Machine learning models (13) and deep learning models(5)	Character based learning	ML: 90–95%, DL: ~96%	Naïve Bayes & QDA performed poorly
Wadhvani et al. (2021)	95,024 names	Multinomial Naïve Bayes & LSTM	TF-IDF, Count Vectorizer, BPE	LSTM ~92%, NB ~72%	Limited generalization across cultures
Ritwik Ghosh et al. (2020)	Multi language dataset	Word Embeddings ,deep learning	Multi-lingual feature learning	Improved over traditional models	Requires large dataset
Pham & Nguyen (2023) – Gendec	Japnese names dataset	Logistic Regression, SVM, NB,RF& mBERT, DistilBERT	Multi-script (romaji, kanji, hiragana), transfer learning	High F1-score (DL > ML)	Complexity of Japanese scripts, homonyms
Panchenko et al. (Russian names)	Russian names dataset	Machine Learning Based Classification	Linguistic & cultural patterns	High accuracy reported	Language-specific limitations
Hu et al.(2021)	Yahoo names dataset	LSTM, BERT, Character based Machine Learning Models	Character Level names Features	Improved gender classification performance using name patterns	Difficult for rare and unisex names
Hallawi et al.(2024)	Iraqi names dataset	Logistic Regression, Naïve Bayes, Random	Name Length, character	MLP and XGBoost achevied accuracy	Unique names, splitting errors reduce

		Forest	features	(96%)	accuracy
Nguyen et al.(2024)	GenderVN 1.0 dataset (Vietnamese names)	Logistic Regression, Naïve Bayes, Random Forest, phoBERT	Transformer-based name representation	PhoBERT achieved 92.3% accuracy	Requires high computational resources
Septiandri (2017)	Indonesian names dataset	Logistic Regression, Naïve Bayes, LSTM, XGBoost	Character embeddings and n-gram features	LSTM achieved accuracy (92.25)	Limited dataset size and Unisexual names
Ghate et al. (2025)	Indian names dataset	CNN, XGBoost, LSTM, GRU	Advanced feature extraction and deep learning	CNN achieved highest accuracy (96%)	Regional diversity and ambiguous names affect prediction

4. Dataset Description

The "Indian and English Names" dataset is used in this study, and it is provided as a two column CSV file (name, gender). The raw file has 125,231 rows. The dataset provides a realistic multicultural baseline for name-based gender classification by merging English origin names (American, British, and other Anglo-Saxon variants) with names of Indian subcontinent origin (Hindi, Bengali, Tamil, Punjabi, and other regional variants).

Category	**Count**	**Percentage (%)**
Female (f)	75,971	60.67%
Male (m)	49,260	39.33%
Total	125,231	100%

Table 1: Class distribution of the Indian and English Names dataset

The modeling dataset, after preprocessing, consists of 108,910 unique name-gender pairs. Class Distribution of dataset after cleaning the data is defined in table 2.Female names are the majority

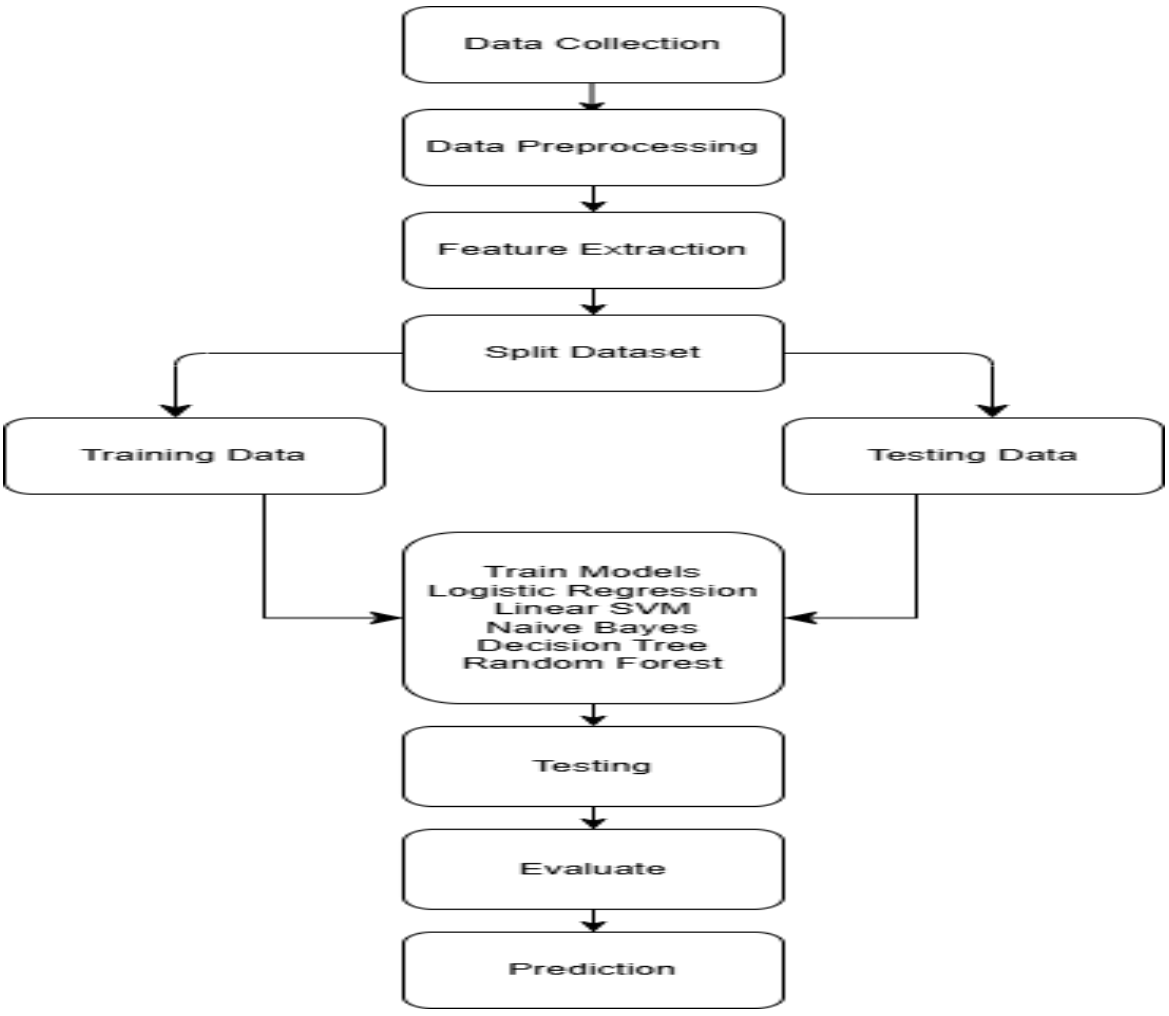
class (61.09%) and male names are the minority class (38.91%), which is a moderate imbalance and is balanced at the training-set level using SMOTE.

Category	**count**	**Percentage (%)**
Female (f)	66,535	61.09%
Male (m)	42,375	38.91%
Total	108,910	100%

Table 2: Class distribution of the cleaned Indian and English Names dataset

5. Methodology

The intended approach for gender prediction using names is based on a machine learning classification approach. The complete workflow contain multiple stages including dataset collection, data understanding, preprocessing, feature extraction, creation of machine learning models, and assessment of performance. The purpose of this methodology is to transform raw data into meaningful insights and develop an efficient model for predicting gender accurately. Figure illustrates the proposed system's whole workflow.



International Journal of Computer Science & Communication (ISSN: 0973-7391)
Volume 17 • Issue 2 pp. 222-233 March 2026-Sept 2026 www.csjournals.com

5.1 Data Collection

The study contains approximately **125,231 name samples** of Indian and English dataset with their corresponding gender labels. The dataset contains two major attributes **name**, **gender**.

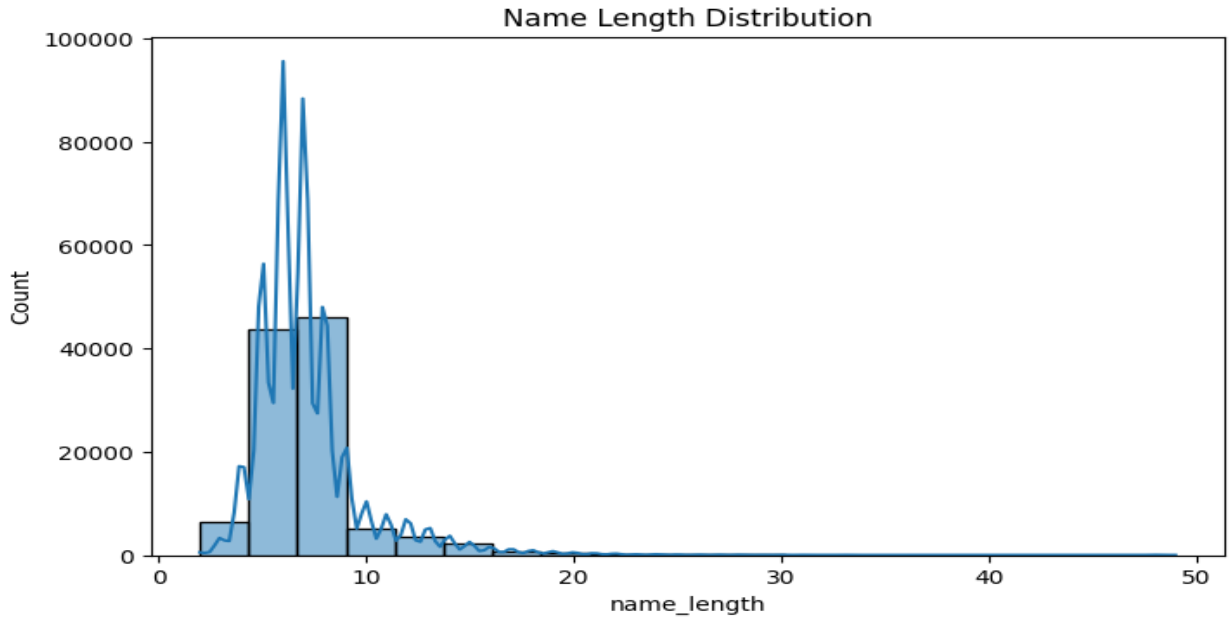
5.2 Data Preprocessing

Data preprocessing is necessary step in machine learning due to raw data may contain inconsistencies, duplicate records, or unnecessary information. The preprocessing stage improves dataset quality and prepares the data for model training.

There are following preprocessing operations are performed on data:

5.3 Exploratory Data Analysis

Before modeling, exploratory analysis is carried out. The distribution of gender visualized as a count plot, confirms the moderate imbalance reported in Table. Name-length analysis, calculated as the character count of each cleaned name, shows a mean length of 7.19 characters and standard deviation 2.72 minimum of 2 characters and a maximum of 49 characters, with the inter-quartile



range extent across 6–8 characters. This relatively short and tightly distributed name length is favorable for character n-gram modeling, since the chosen n-gram range 2–5 characters comfortably extent the majority of each name is structured without trim.

5.4 Missing Value Analysis

Missing values are handled by using isnull() function in machine learning. If any value is missing in dataset then those records are removed because these negative values affect the model performance.

5.5 Duplicate Record Removal

Duplicate records are identified and removed to avoid repeated learning of the same patterns during model training. Removing duplicate names improves dataset consistency, model generalization, Prediction reliability.

5.6 Text Cleaning and Normalization

Since names are textual data, normalization is applied before feature extraction.The following operations are performed conversion of names into lowercase, removal of unnecessary spaces , removal of unwanted characters , standardization of text format.

5.7 Label Encoding

Machine learning algorithms require numerical labels instead of text labels. The gender values are converted as Male = 0 and Female = 1

5.8 TF-IDF-Based Feature Extraction

Following preprocessing, character-level Term Frequency-Inverse Document Frequency is used to convert names into numerical feature vectors. Because names contain significant patterns such as prefixes, suffixes, character combinations, and spelling structures, character-level attributes are used.

The frequency of a character pattern in a name is represented by term frequency.

The significance of that character pattern in the entire dataset is represented by the inverse document frequency.

5.9 Split Train-Test Data

Training and testing sets are separated from the processed dataset. While testing data evaluate the performance of on unseen data, training data is used to identify patterns. For testing 20% data was used while the remaining 80% data was used for training the Model.

5.10 Model Development

Different classification algorithms are applied to compare performance.

Logistic Regression

For binary classification, logistic regression is a supervised classification technique.

Support Vector Machine.

Support Vector Machine is used to identify an ideal decision boundary between the male and female classes. It works effectively in high-dimensional TF-IDF feature spaces.

Naive Bayes

For text categorization, the probability-based Naive Bayes classifier is frequently employed.

Decision Tree

Decision Tree creates classification rules based on extracted name features. It divides data using decision conditions until the final gender class is predicted.

Random Forest

Random Forest is an ensemble learning system that combines numerous decision trees. Prediction accuracy, generalization capacity, and overfitting resistance are all enhanced. Several samples and features are chosen using Random Forest to generate various trees, and their predictions are combined using majority vote.

6 Model Evaluation

Four metrics are used to estimate the performance of trained models such as Accuracy, Precision, Recall, and F1-Score.

TP = True Positive, TN =True Negative, FP =False Positive and FN =False Negative

Accuracy = (TN + TP) / (TN + FP + FN + TP)

Recall = TP / (TP + FN)

Precision = TP / (FP + TP)

F1-Score = 2*(Precision * Recall) / (Precision + Recall)

Prediction Output

After trained and evaluate the model, the final trained model accepts a new person's name as input.

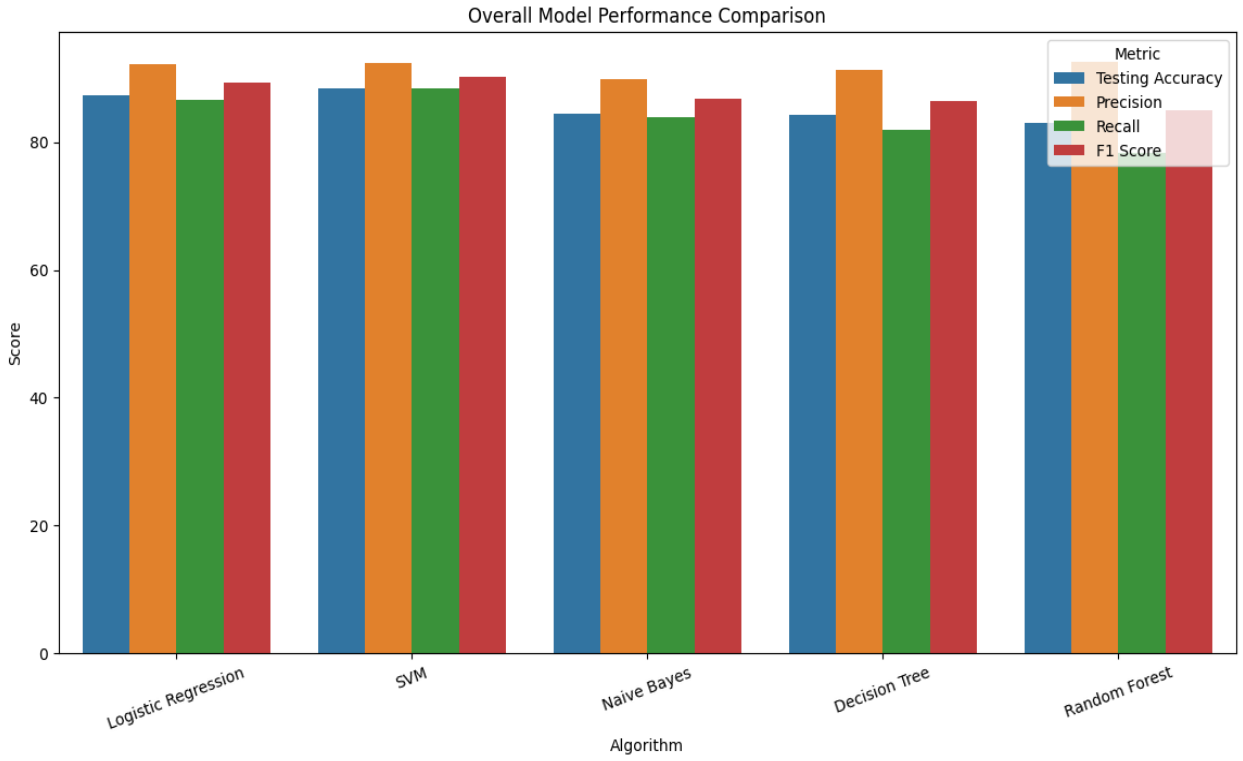
7. Results

7.1 Model Performance Analysis

Table1 reports training accuracy (on the SMOTE-balanced training set), and test accuracy, precision, recall and F1-score on primary, imbalanced 21,782-name test set for all five classifiers. A fixed random seed (42) and the same stratified train-test split and SMOTE-balanced training matrix are used in every experiment.

Model Used	**Train Accuracy (%)**	**Test Accuracy (%)**	**Recall (%)**	**Precision (%)**	**F1- Score (%)**
Logistic Regression	89.55	87.39	86.59	92.29	89.35
Linear SVM (LinearSVC)	91.68	88.42	88.32	92.38	90.31
Multinomial Naïve Bayes	86.08	84.39	83.98	89.80	86.79
Decision Tree	88.82	84.05	81.53	91.43	86.20
Random Forest	86.79	83.02	78.70	92.38	84.99

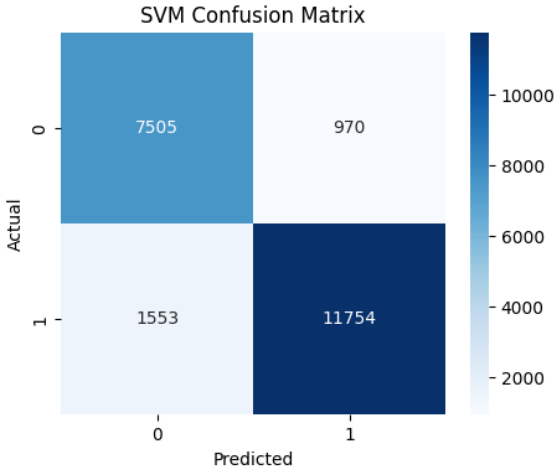
International Journal of Computer Science & Communication (ISSN: 0973-7391)
Volume 17 • Issue 2 pp. 222-233 March 2026-Sept 2026 www.csjournals.com



7.2 Confusion Matrix Summary

Table 2 summarizes the test-set confusion matrices for each model, expressed as true negatives (male correctly predicted), true positives (female accurately predicted), false negatives (female forecasted as male), and false positives (male predicted as female).

Model	**True Male (TN)**	**Male-Female (FP)**	**Female-Male (FN)**	**True Female (TP)**
Logistic Regression	7,512	963	1,784	11,523
Linear SVM	7,505	970	1,553	11,754
Naïve Bayes	7,206	1,269	2,132	11,175
Decision Tree	7,458	1,017	2,458	10,849
Random Forest	7,611	864	2,835	10,472



8. Conclusion and Future Work

8.1 Conclusion

In this research, five traditional machine learning algorithms were compared end-to-end for gender prediction from Indian and English first names, built on a fully reproducible cleaning, feature-engineering, and class-balancing pipeline. After removing missing values and 16,265 duplicate rows, the working dataset comprised 108,910 names (61.09% female, 38.91% male). Names were represented using character level TF-IDF n gram features (n = 2 to 5, max features = 8,000), and the resulting training-set class imbalance was corrected using SMOTE rather than class-weighting. Linear SVM (C = 0.5) achieved the best test-set performance (88.42% accuracy, 90.31% F1-score) and was selected as the final deployed model for single-name gender prediction, just ahead of Logistic Regression, Naive Bayes, Decision Tree, and Random Forest followed the two linear models, with Random Forest showing the largest precision recall gap. These findings are consistent with the broader literature on name-based gender prediction, confirming that selective linear models combined with character n-gram features remain a strong, computationally inexpensive baseline for multicultural name datasets, even when class imbalance is corrected via synthetic oversampling rather than weighted loss functions.

8.2 Future Work

SMOTE based oversampling is directly compared to class weighted versions of the same five models that divide equally in order to measure the precision recall trade-off.

Deep learning models: combining transformer-based models (BERT, DistilBERT, XLMR) with character level LSTM, BiLSTM, CNN, and GRU architectures. Previous research demonstrates that these models can significantly outperform classical methods in terms of accuracy on similar tasks.

Region specific subsets: to facilitate language specific modeling and the study of cross linguistic generalization, Indian names are divided by regional language family (e.g., Tamil, Bengali, Marathi, Telugu, Punjabi).

Extra features: adding phonetic encodings (like Soundex), first/last character distributions, and name length to TF-IDF features.

Multi-model ensembling, which involves reintroducing a majority-voting or stacked ensemble across all five trained models once the performance of each model on SMOTE balanced data has been described.

References & Bibliography

[1]. R. C. B. Rego, V. M. L. Silva, and V. M. Fernandes, "Predicting Gender by First Name Using Character Level Machine Learning," *arXiv preprint arXiv:2106.10156*, Jun. 2021, doi: 10.48550/arXiv.2106.10156.

[2]. R. Ghosh, "Name Based Gender Identification Using Machine Learning and Deep Learning Models," *Research Article*, 2022.

[3]. H. Q. To, N. L. T. Nguyen, K. V. Nguyen, and A. G. T. Nguyen, "Gender Prediction Based on Vietnamese Names with Machine Learning Techniques," in *Proceedings of the 4th International Conference on Natural Language Processing and Information Retrieval (NLPIR)*, Seoul, Republic of Korea, Dec. 2020, pp. 55–60.

[4]. B. A. Wadhvani, S. Jagani, K. Rajani, H. Magatarpara, and D. Chauhan, "Name Based Human Gender Classification Using Machine Learning," *International Journal of Engineering Research and Technology*, 2021.

[5]. D. T. Pham and L. T. Nguyen, "A Machine Learning Based Framework for Gender Detection from Japanese Names," *International Conference on Artificial Intelligence and Data Processing*, 2023.

[6]. M. Panchenko and S. Teterin, "Gender Prediction from Russian Names Using Statistical and Machine Learning Approaches," in *Proceedings of the International Conference on Computational Linguistics and Intelligent Text Processing (CICLing)*, 2014.

[7]. H. Liu and P. Zhang, "Name-Based Gender Classification Using Character-Level Features and Machine Learning Algorithms," *IEEE Access*, vol. 8, pp. 192450–192461, 2020.

[8]. A. M. Rego, J. C. Ferreira, and F. Batista, "Gender Identification from First Names Using Machine Learning Approaches," *Proceedings of Computational Processing of Portuguese Language*, 2021.

[9]. S. Bhardwaj, "Indian and English Names Gender Dataset," *Kaggle Dataset*, 2025.

[10]. T. Joachims, "Text Categorization with Support Vector Machines: Learning with Many Relevant Features," in *Proceedings of the European Conference on Machine Learning*, pp. 137–142, 1998.

[11]. UCI Machine Learning Repository, "Gender by Name Dataset," University of California, Irvine, CA, USA. [Online].