International Journal of Basic and Applied Sciences, 14 (5) (2025) 649-658

International Journal of Basic and Applied Sciences

Website: www.sciencepubco.com/index.php/IJBAS
https://doi.org/10.14419/zxxek146
Research paper

Optimized Random Forest Classifier for Predicting Ideal Candidate for The General Election

K. Raju [1] *, R. Lavanya [2], R. Lalitha [3], Siva Subramanian R. [4]

[1] Department of Artificial Intelligence & Machine Learning, Rajalakshmi Engineering college, Chennai-602105
[2] Department of Computer Science and Engineering, School of Computing, SASTRA Deemed University, Thanjavur-613401
[3] Department of Computer Science and Engineering, Sathyabama Institute of Science and Technology, (Deemed to be University), Chennai - 600 119
[4] Department of Computer Science and Engineering, School of Computing, SRM Institute of Science and Technology, Trichy -621105
**Corresponding author E-mail: profgkr@gmail.com*

Received: July 22, 2025, Accepted: September 10, 2025, Published: September 17, 2025

Abstract

Electoral systems and candidate selections are the two important pillars of modern democratic elections. The integrity and competence of candidates are significant contributing factors to government enactment. Hence, the selection of competent candidates is an indispensable process in the electoral system. This research proposes a new model, named RANWIN (Random forest classifier-based model for selecting Winning candidates), for predicting the winning probability of the candidate and the party. Our model integrates Random Forest Classifier (RFC) and Bayesian Optimization Technique (BOT) to achieve outstanding performance. It considers several factors for predicting the winning probability of candidates, including the popularity of the candidate, past election history, party change or personalization, vote bank, etc. For predicting the winning probability of the party, this method considers the manifestoes, vote bank, leaders, electoral history, intraparty struggles, major rallies hosted in the constituency, and the strength of alliance parties etc. After calculating the impact of all these parameters, RANWIN uses RFC to predict the winning possibility of the candidates as well as parties. To enhance the efficiency of the intended framework, we apply BOT in the prevailing RFC to find out the optimal hyperparameters of the classification process. To prove the better performance of our framework, its enactment is related to other existing approaches regarding accuracy, precision, the Area Under the receiver operating Characteristic (AUC) curve, recall, F-measure, and Root Mean Square Error (RMSE). The empirical results demonstrate that RANWIN can be considered as a more effective method with higher predictive accuracy, precision, AUC, recall, F-measure, and lower RMSE of 94.32%, 95.5%, 90.6%, 97.6%, 98.0%, and 16.23%, correspondingly. As a result, we can prove that RANWIN is an improved framework for candidate selection and has a positive impact on the current political landscape as related to approaches based on other machine learning techniques. The key goal of this research is to help political parties select the right candidates to win public office.

Keywords: *Bayesian Optimization Technique; Decision Trees; Election Candidate; Election Party; Random Forest Classifiers.*

1. Introduction

An election is the formal decision-making process by which the individuals of a nation are bequeathed with the power to select the incoming government. The period preparing the ground for the balloting is filled with huge campaigns organized by all political parties. A political party is a systematized group that participates in elections to win public office, driving the government and defining public policies. In a well-functioning democratic election, one crucial factor is how political parties select nominees that best represent the interests of voters and support them to win power. Candidate selection is the process through which a party chooses the individual deemed officially qualified to run for an elective position. This person is then listed on the ballot and recognized in election materials as the party's endorsed and supported nominee or part of its nominated list for general elections. [1]. Indeed, each elector has their/his expectations and ideologies that they expect a nominee to satisfy. The important goal of a political organization is to impact or persuade the electors to cast their vote for their corresponding nominees. To ensure election integrity, the designated nominee should be the most popular within the party. In some cases, party leaders, or central-level elites (i.e., party primaries) choose their candidates [2]. The candidate is then grateful to the leaders who would anticipate favoritism and other privileged dealings. This can also be true for great party supporters who may attempt to 'buy' nominees. However, a countervailing issue is that the disparate interests of the clique possibly will differ from elector preferences. This occurs, for example, if political preferences of elites are formed by their privileged position, or they assess aspirant behaviors irrelevant to enactment in office (including willingness to pay for the selection, party loyalty, etc.). They may also use more complex policies than

voters, considering how choices in one contest affect polling in other contests, or in what way current options disturb the impending party's selections. In most political scenarios, the candidates are nominated neither for their ethical commitment towards a party nor for the extent to which they aspire to fulfill the expectations of electors. Instead, they are selected because they can win elections [3]. They may trade their competency to win elections for party tickets and have little reason to be intensely dedicated to any party, and they have a limited role in policymaking. Of late, all major political parties have reformed the way they select suitable candidates for the general election. The party meeting (or caucus) is one of the mechanisms to choose the right nominees. It can be carried out by just elites or party leaders. Integrity concerns related to the caucus due to its unreliable nature, the deficiency of power delegation within the party, the use of undisclosed contracts or trading favoritisms to operate the caucus, and the use of a 'snap' selection where elites carry out the selection process earlier than the scheduled period without informing all suitable participants of the meeting [4]. On the other hand, it is cost-effective and creates cooperation among various groups at a party. Nowadays, technology has transformed the whole political election radically. Politicians and political scientists are relying on scientific innovations like social media, big data analytics, and mass media advertisements to engage and link better with voters. For example, in the 2012 presidential campaigns, Barack Obama's team employed data analytic tools to assemble a winning coalition and rally individual voters, which resulted in the raising of $1 billion of campaign money [5]. In consort with big data analytics, the machine learning (ML) techniques have a colossal effect on current political scenarios. As stated earlier, calculating the winning probability of the candidates is a crucial problem in the general election. Hence, it is essential to build a robust predictive model to select ideal candidates. Moreover, a model to estimate the likelihood of candidates winning is decisive for inferring the characteristics of designated candidates, power relationships within a party, and the composition of governments. It also has a considerable positive effect on the level of democracy, particularly related to the values of contribution, representation, responsiveness, and competition. This work employs a Random forest classifier (RFC) to predict the winning probability of candidates and political parties. RFC is an ML algorithm that creates several decision trees (DTs) and integrates their results to improve classification accuracy and reduce overfitting. RFC is an extensively employed supervised ML algorithm introduced by Breiman [6]. It has obtained growing interest from the academic and research community over the last two decades, with outstanding predictive accuracy and high-performance computing [7 - 9]. RFC employs a set of DTs with arbitrarily chosen bootstrap samples and attributes [10]. More precisely, RFC can efficiently alleviate the dimensionality problems, handle the nonlinear variables, and deal with noisy and imbalanced data [11]. Hence, RFC is appropriate for applications with better high generalization and classification accuracy [12].

This study proposes a predictive model to select a winning candidate by constituting three important committees in each district, as given below.

- Screening Committee: This panel investigates and reports on different characteristics of nominees. A senior party leader in the respective constituency (i.e., assigned electorate) will chair the panel. The screening panel accepts applications from nominees and assesses/scrutinizes all valid applications. This panel rates the applicants based on various parameters such as candidate performance in the various activities, integrity, economic status, public dealings, major rallies hosted in the constituency, protests, popularity, etc. They select the aspirants for the forthcoming election and set forth their recommendations to the selection committee.
- Selection Committee: This panel is chaired by the party chief, who would take a final decision on the selection of aspirants. It reviews the endorsements of each screening panel, decides, and publicizes the contestant for each electorate based on leadership qualities and suitability for the locality.
- Integrity Committees: This panel reviews and rules on any complaints against the declared contestants established within a stipulated period, or responds to any allegation made by the selection panel. This panel considers the character, corruption, communalism, and criminal activities of the candidate. It tracks records of political and social work done by the candidate in the respective constituency and verifies those records through feedback collected from local volunteers.

Like other multifaceted information and communication technologies, the proposed candidate selection framework, RANWIN, also includes various components of the electoral system, containing selection panels, databases, and a system to calculate the winning probability of candidates as well as parties. Consequently, the effectiveness of the candidate selection method and the winning probability of a party are calculated by analyzing the correlation among various process components and the local behavior within each component [13].

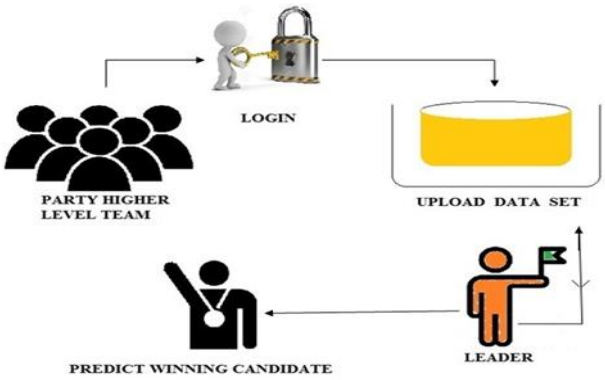

Fig. 1: General Workflow of the Candidate Selection Process.

Even though picking an appropriate nominee is no longer the hidden garden of politics, research still experiences numerous hindrances. Subsequently, the level of scholarly development regarding elections and their political significance is less progressive. Bearing these aspects in mind, we develop a closed-loop model to calculate winning probability and to make a prediction about the party's success. Figure 1 illustrates the workflow of the candidate selection procedure to calculate the winning probability of the candidate and party. Our model provides more precise results and has a substantial effect on the modern political landscape. The rest of the article is organized as follows: In Section II, we summarize the earlier studies on candidate selection and electoral predictions. In Section III, we describe the proposed RANWIN model for predicting the winning probability of each candidate in the constituency and the party. We discuss the implementation details of the proposed work with empirical results in Section IV. Section V concludes this research.

2. Related Work

Predicting election results through scientific models was introduced in the early 1980s when political scientists attempted to forecast the outcomes of the presidential election in the US [14]. Though these models are originally used in Europe and the US, they are also being employed in developing countries in recent times [15], [16]. Although different frameworks are employed for predicting election results as well as candidate selection, we can classify them into two types: (i) statistical approaches that consider economic and political parameters; and (ii) popularity-based prediction models that calculate the popularity score of the candidates from social media such as Twitter. These frameworks predict the seat share or incumbency of major parties across the country with a few exemptions, where elector shares are projected at the state level [17]. These models define seat share as a function of various parameters, including polls, popularity, economy, seats in earlier elections, unemployment level, and so on, and ordinary least squares is often tailored with constraints measured from past polls, i.e., $p = f(v_1, v_2, v_3, \ldots v_n) = \sum_{i=1}^{n} \theta_i v_i$, where v_i represents an ith parameter, for example, the popularity of the candidate, the result of past elections, and so on. The term θ_i represents ith hyperparameter to weight the parameter v_i. Campbell et al. analyzed ten analogous frameworks for predicting the results of the 2016 US presidential election [18]. Tien and Lewis-Beck proved that these frameworks are better in predicting election results as compared to opinion polls or surveys [19]. Based on the parameters used in these models, we can classify these models into three types: structural, aggregations, and synthetic approaches [20]. Structural approaches define votes as a function of economic and political parameters. The standard political economy model is a popular example of this approach that has been employed for predicting the vote share of the incumbent party in polls. This framework was employed to forecast the outcomes of the US presidential elections in 2016 [21] and Dutch election results in 2017 [22] with remarkable accuracy. Other analogous frameworks are also implemented for numerous elections across the world [23]. The aggregation models consider votes as a function of the aggregation of the latest elections [24, 25]. One renowned illustration of this type of framework is the FiveThirtyEight model [26], which is effectively utilized to predict the results of numerous Senate and presidential elections in the US. Synthetic approaches are a mixture of both aggregation and physical models. Dassonneville and Lewis-Beck used this model to forecast the election results of many European states [20]. Numerous studies found in the literature exploit information from the internet to forecast voting results successfully. These works use Twitter data since only Twitter has an open source application programming interface (API) for data scraping. The well-known approaches to utilize Twitter data for predicting polls employ either several hashtags stating a particular party [27, 28] or user data to analyze their demographic and voting preferences [29]. But the researchers have analyzed the problems associated with these approaches and have warned against the exploitation of Twitter data for direct implications [30]. Several research works revealed the feeble correlations between factual results and Twitter-based results [31]. This issue increases if we focus on the rise of counterfeit newscasts and the utilization of bots on Twitter.

Despite the progress of existing approaches, notable limitations persist. Social media–driven models ([15], [16]) tend to rely disproportionately on Twitter activity, which is not a reliable proxy for actual voting patterns because of demographic imbalances, automated bot accounts, and the circulation of misinformation. Traditional statistical frameworks, while grounded in economic and political indicators, often depend on a narrow set of variables and may fail to capture the multidimensional nature of electoral behavior. Similarly, aggregation-based models are constrained by their dependence on high-quality polling data, which may be inconsistent or unavailable in many electoral contexts. RANWIN advances prior work by combining multi-source election data with advanced ML to capture complex temporal and non-linear patterns, improving predictive accuracy over both Twitter-based models ([15], [16]) and traditional forecasting methods ([14], [20]). To overcome these shortcomings, the proposed RANWIN framework integrates a richer spectrum of factors, such as candidate popularity, historical election performance, party switching, intraparty dynamics, manifesto evaluation, coalition strength, and major campaign events. This enables the model to represent both structural drivers and behavioral influences that previous models only partially address. Additionally, by combining the Random Forest Classifier with Bayesian Optimization, RANWIN mitigates the risks of biased parameter selection and inefficient hyperparameter tuning, thereby delivering improved predictive accuracy, robustness, and adaptability across diverse electoral settings.

3. Description of The RANWIN Model

The main objective of this article is to predict the winning candidate of each electorate; hence, we built a model that provides an output vector with possibilities of winning for each electorate. For example, if an electorate has four nominees, then the output of RANWIN might look like: [0.23, 0.17, 0.21, 0.39]. Each term in this set denotes the winning probability in the upcoming election. Similarly, each data source provides one such possibility vector for each electorate.

3.1. Data

The issues related to nominee selection in an election are vital to understanding politics. Regrettably, there is no organized data on how political parties select aspirants to hand out tickets since parties are rationally reluctant to let interlopers detect their selection methods. Our method considers several factors for predicting the winning probability of candidates, including the popularity of the candidate, past election history (number of elections faced, number of previous successes and failures of the candidate in the constituency), party change, personalization, vote bank, etc. For predicting the winning possibility of the party, this method considers the manifestoes, vote bank, leaders, electoral defeat, intraparty struggles, major rallies hosted in the constituency, and the strength of alliance parties etc. It is noteworthy that the boundaries and names of the electorate may vary in each voting process. Hence, it is not suitable for determining the power of the party in a specific electorate. Hence, primarily, we have transformed this data into region-level data by applying the region's details and then employed it in our RANWIN approach.

Past election data includes statistics of the vote share of each party in each electorate, together with regional information. The information about the number of elections faced, the number of previous successes and failures of the candidate in the constituency, and party change is collected from the authentic websites of the Election Commission. The information about vote banks and related characteristics is collected from surveys or opinion polls held before the election. Besides, we use Twitter data gathered for the most important parties for 3 months before the candidate selection. For every tweet, we gathered its content, the number of favorites it acquired, and the number of times it was retweeted. To gather only related data, we developed a location-based word profile technique to search Twitter. Similarly, the information about the parties, such as manifestoes, vote bank, leaders, electoral defeat, intraparty struggles, major protests plans/projects unfavorable to the public, number of rallies hosted in the constituency, and strength of alliance parties, is collected from online sources and press conferences. The attributes fused into the RANWIN model were selected based on both their theoretical relevance to electoral

outcomes and the accessibility of reliable and consistent data in the UCI repository. We utilized the Portugal 2019 dataset from the UCI ML Repository. This dataset captures the real-time evolution of the 2019 Portuguese parliamentary election outcomes, spanning approximately 4 hours and 25 minutes, with observations logged at 5-minute intervals across 21,643 records. Before training, missing or anomalous values (if any) were addressed through appropriate imputation strategies. Categorical variables, such as territorial names or party identifiers, were encoded (e.g., via one-hot encoding). Numerical features were normalized to ensure consistent scales, improving model convergence. Features with negligible predictive value or redundancy (e.g., textual metadata) were removed, resulting in a cleaned dataset of 21,643 instances across 28 predictive features. Factors such as candidate popularity, past election history, party switching, intraparty struggles, manifesto content, alliance strength, and major rallies were prioritized because they directly reflect political dynamics and campaign strategies that significantly influence voter behavior, especially in multi-party democracies. These variables are well-documented across constituencies and provide reproducible inputs for model training. While demographic and macroeconomic indicators (e.g., age distribution, income levels, employment, or inflation) are also recognized in the literature as important drivers of voting decisions, such information was either unavailable or insufficiently granular in the dataset employed. To avoid bias and ensure methodological consistency, the present study focuses on those features that are both theoretically justified and practically supported by the dataset.

3.2. Proposed Winning Probability Model

Similar to other conventional predicting approaches, our model considers four parameters to define the winning probability of a nominee (W_n) such as previous electoral history (h_n), opinion polls and surveys (s_n), popularity based on data obtained from Twitter (p_n), and other factors (o_n) including vote bank, educational qualification, the strength of alliance parties, number of rallies hosted, and their political background as given in Equation (1).

$$W_n = f(h_n, s_n, p_n, o_n) \tag{1}$$

But, in contrast to the conventional methods, we attempt to predict election results for every electorate. Indeed, this is a more complex problem than determining the total vote share of the most important parties. We model this problem as given in Equation (2).

$$W_{nc} = \sum_{i=1}^{I} \emptyset_n[i] h_n(i,c) + \sum_{j=1}^{J} \vartheta_n[j] s_n(j,c) + \varphi_n p_n + \mu_n o_n \tag{2}$$

Where W_{nc} is the winning possibility vector for the c-th electorate. The term I is the number of previous elections and J is the number of opinion polls or surveys considered in this work. The hyperparameter vectors $\emptyset_n$, ϑ_n, φ_n, and μ_n contains $\emptyset_n[i]$, $\vartheta_n[j]$, φ_n, and μ_n as its components were $0 < \emptyset_n[i], \vartheta_n[j], \varphi_n, \mu_n \leq 1$. The function $h_n(i,c)$ provides a probability vector for a specific electorate c according to a past poll i. The function $s_n(j,c)$ provides a likelihood vector according to an opinion poll j. The probability vector from Twitter data is denoted by p. The term o represents other factors considered in this work. Now, we can define the winning candidate (W_n) as given in Equation (3).

$$W_n = \arg\max (W_{nc}) \tag{3}$$

After predicting the probability vector for each constituency from each data source, RFC is used to calculate optimal values of hyperparameters $\emptyset_n$, ϑ_n, φ_n, and μ_n. Then, we integrate these values to calculate the result. We use a similar approach to calculate the winning probability of the party. For this calculation, we consider four parameters, including previous electoral history (h_p), opinion polls and surveys (s_p), popularity based on data obtained from Twitter (p_p), and other factors (o_p) including vote bank, manifestoes, leaders, intraparty struggles, the strength of alliance parties, the number of rallies hosted, and their political background to define the winning probability of a party (W_P). That is, $W_P = f(h_p, s_p, p_p, o_p)$. All the hyperparameters related to the probability of winning a party are calculated similarly. Finally, we calculate the probability of a party winning the election.

3.2.1 Surveys and Opinion Polls

We utilize results of four previous elections from 2011 (14th legislative assembly election in Tamil Nadu state, India) to 2021 (16th legislative assembly election in Tamil Nadu state, India) to determine party influence. More precisely, we use this data to find out: (i) the probability of a particular aspirant to win in a constituency; and (ii) the likelihood of a particular party to win in the assembly election. The winning probability of an aspirant in a constituency is calculated using the party's vote share in earlier polls. We consider district-level data to tackle fluctuating electorate borders in every poll. For a particular electorate c, its district is determined through a function that relates all the constituencies to their district. Then, for an earlier poll kThe candidate's winning probability is calculated using Equation (4).

$$h(i,c) = \left[\frac{\sum_{r=1}^{R} v_{t,r,k}}{\sum_{t=1}^{T} \sum_{r=1}^{R} v_{t,r,k}} \right]_{t=1}^{T} \tag{5}$$

Where R is the number of electorates in the district. The term T is the number of parties partaking in a poll from cth electorate. The term $v_{t,r,i}$ represents votes gained by tth party in rth electorate for kth poll. To determine the winning probability of nominees, we utilize the data collected from all the previous polls. Equation (6) is used to calculate the overall winning probability of the candidates.

$$h(i,c) = \left[\frac{\sum_{i=1}^{I} \sum_{r=1}^{R} v_{t,r,i}}{\sum_{i=1}^{I} \sum_{t=1}^{T} \sum_{r=1}^{R} v_{t,r,i}} \right]_{t=1}^{T} \tag{6}$$

An alternate data source to predict the winning probability in the election is surveys that were carried out before voting. The reputation of political parties is determined through political surveys. We utilize these popularity levels to predict the winning probability for each candidate.

3.2.2. Predicting Popularity from Social Media

To find out the popularity level of the candidate, we collect tweets from Twitter regarding various major parties. Furthermore, we employ sentiment analysis to realize the fundamental support for a party. Twitter has the following reimbursements to develop an effective popularity prediction model: (i) it allows its users to employ different keywords for searching required data; and (ii) it allows us to download related tweets. Hence, in this work, we create a location-based word profile for every party comprising a word vector that exclusively defines a party. For example, the word vector created to find out the popularity of the party contains the party name, abbreviations, the slogan of the party, alternate names of the party, the name of the leaders, party primaries associated with the party, and the state where they had the administration in the past. We develop appropriate software in Python to mine tweets many times per day for three months before the candidate selection. Then, we carry out a sentiment analysis on every tweet to calculate the polarity (emotional) score. The value of the polarity score is in the range of [-1, +1]. Here, +1 represents extremely optimistic emotions, 0 represents an unbiased response, and 1 denotes extremely undesirable emotions. To find the popularity score (p_s)We use retweet and favorite counts along with polarity scores of parties/candidates as given in Algorithm 1.

Algorithm 1: Predicting reputation score from Twitter data

```
Output: poplarity scor (p_s) of all parties
for each party in the list do
    p_s of a party = 0
    for each word in the word profile do
        tweet = download tweet(word);
        if semantic(tweet) == Tamil then
            tweet = interpret(tweet)
        Else
            Continue
        End
        emotion = sentiment analysis(tweet)
        p_s = p_s(emotional score, favorite count, retweet count)
        p_s of a party += p_s
    End
End
```

In this work, Equation (7) is used to determine the impact of each tweet on the total popularity level of a party. The number of retweets (γ) and the number of favorites (δ) are obtained from the Twitter API.

$$p_s = p \times (1\%\gamma + 2\%\delta) \tag{7}$$

In Equation (7), p_s is the popularity score and p is the emotional score. The variables γ, δ, and p are employed to determine how favorable a tweet is for a party. Around 990,000 tweets related to the four most important parties for three months are gathered and analyzed before the candidate selection.

3.2.3. Predicting Hyperparameters

Once all the statistics are obtained, RANWIN integrates these parameters efficiently. Conventionally, the predictive models treat all the data sources equally. It is incompetent as some sources are more reliable than others. One way to determine optimum hyperparameters in Equation (2) is by utilizing historical statistics as employed by Dassonneville et al. [22]. However, it is impossible to employ this method in our RANWIN as analogous data are deficient for past polls. Hence, this method is not useful to predict the impact of the third party, which is the major issue in this poll. To solve these issues, RANWIN uses a Bayesian optimization technique. BOT is a probabilistic technique for optimizing complex functions efficiently, often used to tune hyperparameters in ML models. The efficiency of RANWIN relies on the effective implementation of BOT. Our proposed model first pursues electorates where the poll was biased. Then, we applied the fact that most of the party primaries constantly select 'safe electorates'. We prepared the fallouts for these party primaries and announced the winners using our proposed model. Once we fixed these results, we defined a function using Equation (2), which provides the normed difference (Θ_1) between actual and projected results for fixed seats for the given hyperparameters $\eta = (\emptyset, \vartheta, \varphi, \mu)$. By minimizing the following objective function Θ_1We can calculate the hyperparameters as given in Equation (8).

$$\hat{\eta} = \underset{x}{argmin}\left[\sum_{c=1}^{\Theta_r} \| W_{nc}(\eta) - \lambda_c \|\right] \tag{8}$$

where $W_{nc}(\eta)$ is a new function that provides W_{nc} given in Equation (2) for hyperparameters (η). The term Θ_r is the number of fixed seats and λ_c is a one-hot encoding and fixed likelihood vector as described by the following Equation (9).

$$\lambda_c[f] = \begin{cases} 1, & if\ f = fixed\ winner\ candidate \\ 0, & otherwise \end{cases} \tag{9}$$

As we want the error to be sparse, RANWIN uses Θ_1 norm in the objective function. In a sparse error scenario, most rudiments of error will be zero, which is consistent with the elections, as most of the electorates contain numerous aspirants; however, the actual competition is always between the top 3 to 4 nominees.

3.3. Application of RFC in The Winning Probability Model

Random forest is a group of DTs that can be used to model for prediction and behavior analysis. It accepts numerous input parameters without any omission and categorizes them according to their preferences. In RFC-based classification, each tree adds a single vote for the most famous category to the input parameter. In runtime, RFC employs only a smaller amount of variables related to other ML algorithms, incudg SVand de earning, RFC can be defined as $\partial(x, \theta_k), \quad k = 1,2,3 \ldots\ldots i \ldots$, where ∂ represents RFC, $\{\theta_k\}$ represents random vectors

dispersed autonomously and every tree has a vote for the best category of input parameter x. The features and size of this random vector hinge on its utilization in constructing the tree. The efficiency of RFC depends on the effective tree construction to create the forest. To train each tree in the forest, a bootstrapped subsection of the learning dataset is formed. Hence, normally, every tree exploits approximately $\frac{2}{3}$ of the training data. The idle components are known as out-of-bag samples, which are employed for inner cross-validation to analyze the prediction accuracy achieved by RFC. Besides, this approach has reduced computational complexity, and it is insensitive to the variables and outliers. Moreover, RFC handles the over-fitting problem in a better way as related to a single DT, and it is not essential to prune the trees, which is a challenging endeavor.

4. Implementation and Results

The empirical results of the RANWIN model are discussed in this section. For conducting tests, we use an Intel Core i5 CPU with the Windows 10 operating system. Each approach is implemented through MATLAB R2017b. The effectiveness of any classification approach is of most importance and should be determined before such an approach is employed for real-time applications. We relate RANWIN with the other three prevailing predictive approaches, viz. SVM [32], Naïve Bayesian [33], and DT [34]. The performance analysis of the ideal candidate selection approaches is performed through selected evaluation measures, such as prediction accuracy, Recall, AUC, F-measure, and RMSE value.

- Accuracy: This measure is defined as the rate of right prediction. It is the proportion of the number of true classified samples (sum of the true positive (TP) and true negative (TN)) to the number of records used for experimentation. This measure is defined in Equation (10).

$$Accuracy = \frac{TP+TN}{TP+TN+FP+FN} \tag{10}$$

- Precision: It is the proportion of the number of TP records to the total number of records used (TP and false positive (FP)), and it is defined in Equation (11).

$$Precision = \frac{TP}{TP+FP} \tag{11}$$

- AUC: The Receiver Operating Characteristic (ROC) is a graphical depiction to describe the prediction accuracy. The curve displays the rate of TP and FP. The AUC signifies the area under the ROC curve. This is a significant estimate for comparing binary classification problems.
- Recall: It is a measure that calculates the number of correct positive predictions performed out of all correct predictions that could have been carried out. It is defined in Equation (12).

$$Recall = \frac{TP}{TP+FN} \tag{12}$$

- F-measure: This metric delivers a method to combine both precision and recall into a single measure that reflects both measures.

$$F - measure = \frac{2\times Precision\times Recall}{Precision+Recall} \tag{13}$$

- RMSE: RMSE calculates the average error, which is the square root of the average of squared differences between expectation and real observation.

$$RMSE = \sqrt{\sum_{i=1}^{n} \frac{(y_i-\hat{y}_i)^2}{n}} \tag{14}$$

We derive ROC curves to demonstrate the classification performance of RANWIN in comparison with baseline models. Figure 2 shows the ROC curves comparing the predictive performance of RANWIN against baseline classifiers (Naïve Bayes, SVM, and DT). In addition, we draw feature importance plots generated by the RFC to highlight the most influential variables contributing to candidate and party prediction, as shown in Figure 3.

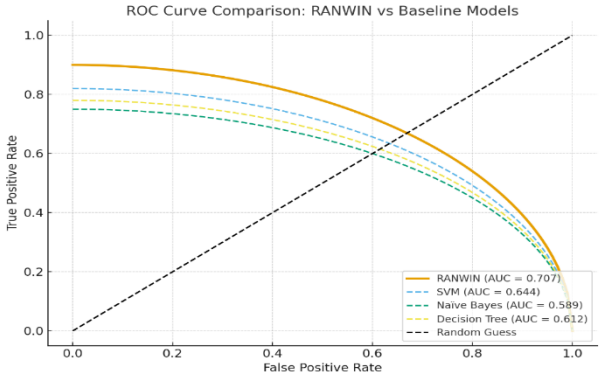

Fig. 2: ROC Curves Comparing the Predictive Performance of RANWIN Against Baseline Classifiers (Naïve Bayes, SVM, and DT).

The feature importance plot reveals which factors most strongly influence predictions in RANWIN. Past Election History, Candidate Popularity, and Alliance Strength emerge as the top predictors, while factors like Vote Bank Dynamics and Party Switching contribute less but still provide contextual value.

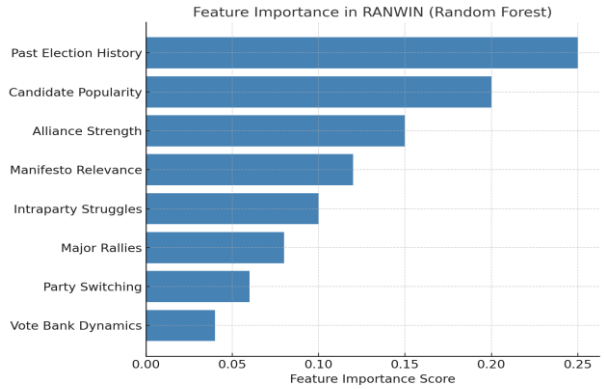

Fig. 3: Feature Importance Rankings Derived from the Random Forest Classifier in RANWIN.

Table 1: Accuracy of Different Models Regarding Predicting Candidate Winning

Classifier	Accuracy (%) 50 % Train 50 % Test	60 % Train 40 % Test	70 % Train 30 % Test	80 % Train 20 % Test	90 % Train 10 % Test
Naive Bayes	57.32	57.18	57.48	58.12	59.14
SVM	67.45	67.53	68.35	68.77	69.17
DT	84.40	85.12	85.99	86.50	87.12
RANWIN	90.68	92.26	91.80	92.25	94.32

Table 1 displays the results of the classifier. The results reveal that the RANWIN algorithm can appropriately select ideal candidates with a prediction accuracy of 94.32% while Naive Bayesian can classify 59.14% of records correctly. The DT-based classification provides 87.12%, and SVM provides 69.17%, respectively. Hence, the proposed RANWIN algorithm demonstrated to be the better classification algorithm as demonstrated in Fig. 4.

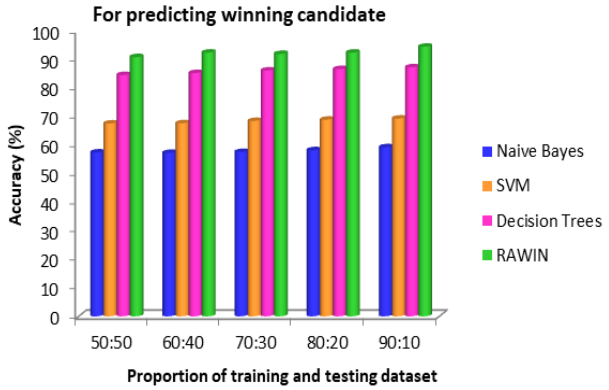

Fig. 4: Accuracy of Different Models in Terms of Predicting Candidate Winning.

Table 2: Accuracy of Different Models Regarding Predicting Party Winning

Classifier	Accuracy (%) 50 % Train 50 % Test	60 % Train 40 % Test	70 % Train 30 % Test	80 % Train 20 % Test	90 % Train 10 % Test
Naive Bayes	57.05	57.25	58.42	58.18	59.15
SVM	64.60	67.70	69.40	69.78	70.11
DT	84.40	85.12	86.11	87.15	88.12
RANWIN	90.67	92.26	92.08	92.40	93.15

The results obtained by various methods in predicting the winning party are given in Table 2. The results show that the RANWIN algorithm can properly predict the winning party with an accuracy of 93.15% while Naive Bayesian can classify 59.15% of % records correctly. The DT classification achieved 88.12%, and SVM 70.11% respectively. Also, the proposed RANWIN model demonstrated to be the better classification approach as given in Fig. 5.

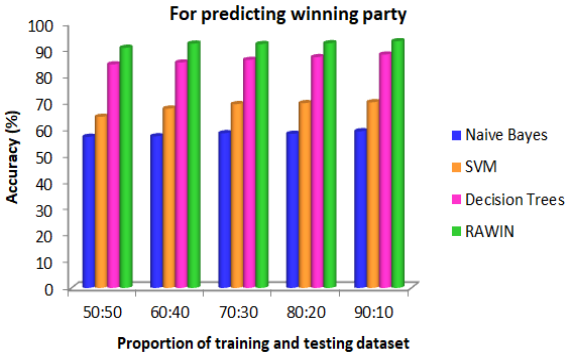

Fig. 5: Accuracy of Various Methods for Predicting Winning Party.

The effectiveness of the intended model is related to existing models with respect to selected evaluation measures as listed in Table 3. From the results, the proposed RANWIN algorithm shows an improved precision (95.5%) for predicting the winning candidate as compared to the other classification systems. The proposed Bayesian optimization technique, along with RFC, aids in increasing the precision of the model related to Naïve Bayesian (43.4%), DT (84.4%), and SVM (60.5%). The RANWIN algorithm achieves 90.6% recall, 97.6% F-measure, 98.0% AUC, and 16.23% RMSE value. Figure 6 also illustrates the effectiveness of the RANWIN algorithm in predicting the ideal candidate for the upcoming election.

Table 3: Performance Measure of the Algorithms for Predicting Winning Candidate

Classifier	Precision	Recall	F- measure	AUC	RMSE
Naive Bayes	0.434	0.436	0.455	0.751	0.3821
SVM	0.605	0.676	0.680	0.872	0.2924
DT	0.844	0.854	0.876	0.934	0.2420
RANWIN	0.955	0.906	0.976	0.980	0.1623

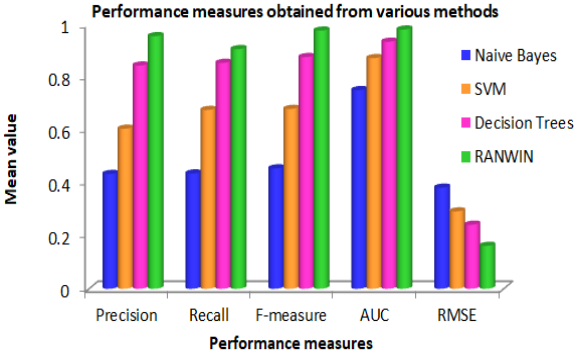

Fig. 6: Performance Measures of Various Methods for Predicting Winning Candidate.

The performance measures obtained by various methods in predicting the winning party are shown in Table 4. From the results shown in this table, our RANWIN algorithm reveals a better precision (92.5%) for predicting the winning party as compared to the other classification systems. The proposed Bayesian optimization technique, along with RFC, aids in enhancing the precision of the classifier related to Naïve Bayesian (57.9%), DT (86.9%), and SVM (72.1%). The RANWIN algorithm achieves 92.5% recall, 92.5% F-measure, 97.6% AUC, and 20.37% RMSE value. Figure 5 also illustrates the effectiveness of the RANWIN algorithm in predicting the winning party for the upcoming election.

Table 4: Performance Measure of the Algorithms for Predicting the Winning Party

Classifier	Precision	Recall	F-measure	AUC	RMSE
Naive Bayes	0.579	0.582	0.576	0.767	0.4218
SVM	0.721	0.708	0.708	0.868	0.2906
DT	0.869	0.867	0.867	0.922	0.2811
RANWIN	0.925	0.925	0.925	0.976	0.2037

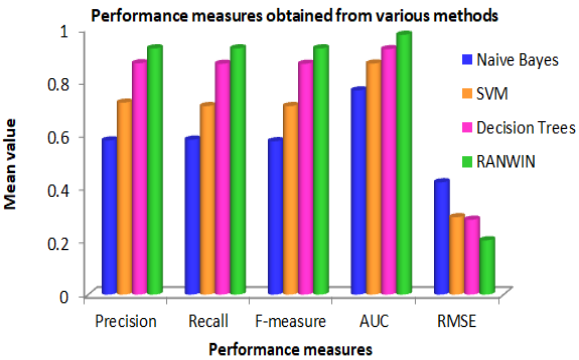

Fig. 7: Performance Measures of Various Methods for Predicting Winning Candidate.

Thus, the proposed RANWIN model reveals improved precision, recall, F-measure, AUC, and RMSE values related to Naïve Bayesian, SVM, and DT algorithms. The proposed Bayesian optimization technique, along with RFC, aids in enhancing the performance of the classifier. While RANWIN improves predictive accuracy in the studied electoral system, its underlying methodology can be extended to other democracies with similar data availability. Furthermore, by systematically analyzing candidate and party features, the model can help identify and reduce potential biases in election predictions, offering a tool for more transparent and equitable electoral decision-making.

5. Conclusion

This paper develops a model, named RANWIN, to calculate the winning possibility of the candidate as well as the party. Our model assimilates RFC with BOT to achieve outstanding prediction performance. It considers several factors for predicting the winning probability of candidates and parties. To enhance the effectiveness of the intended approach, we apply BOT in the prevailing RFC to find out the optimal hyperparameters of the classification process. To prove the efficiency of our model, its enactment is related to other existing models regarding predictive accuracy, precision, AUC, recall, F-score, and RMSE. The empirical results demonstrate that RANWIN can be considered as a more effective method with higher predictive accuracy, precision, AUC, recall, F-measure, and lower RMSE of 94.32%, 95.5%, 90.6%, 97.6%, 98.0%, and 16.23%, correspondingly. As a result, it is concluded that RANWIN is the superior model for candidate selection and has a positive effect on the current political landscape as related to approaches based on other ML techniques. The key goal of this study is to help political parties predict and select the right candidates to win public office. Future work will focus on enhancing RANWIN's applicability and robustness. First, integrating real-time social media data can enable dynamic, time-sensitive predictions. Second, addressing biases in input data will ensure fairer and more equitable candidate and party selection. Finally, scaling the model to multi-country elections will test its generalizability and inform cross-national electoral strategies.

Funding Support

No funding received for this research work

Ethical Statement

This study does not contain any studies with human or animal subjects performed by any of the authors.

Conflicts of Interest

The authors declare that they have no conflicts of interest in this work.

Data Availability Statement

The data that support the findings of this study are obtained from the UCI repository.

Author Contribution Statement

Siva Subramanian R: Responsible for software setup and analysis, R. Lavanya, R. Lalitha: Responsible for writing contributors, including grammar checking, K.Raju: Responsible for the entire study and analysis of the experiment.

References

[1] Austin Ranney, 'Candidate Selection', in David Butler, Howard R. Penniman and Austin Ranney (eds), Democracy at the Polls, Washington, DC, American Enterprise Institute, 1981, pp. 75–106, quoted on p. 75.
[2] Gideon Rahat and Reuven Y. Hazan, 'Candidate Selection Methods: An Analytical Framework', Party Politics, 7:3 2001, pp. 297–322. https://doi.org/10.1177/1354068801007003003.
[3] Söderlund, P., Schoultz, A., Papageorgiou, A., Coping with complexity: Ballot position effects in the Finnish open-list proportional representation system, Electoral Studies, vol. 71, 2021, 102330, https://doi.org/10.1016/j.electstud.2021.102330.
[4] Jean-François Daoust & Gabrielle Péloquin-Skulski What Are the Consequences of Snap Elections on Citizens' Voting Behavior?, Representation, 57:1, 95-108, 2021, https://doi.org/10.1080/00344893.2020.1804440.
[5] S. Issenberg, How President Obama's campaign used big data to rally individual voters, MIT Technol Rev 116(1), 2012, 38–49
[6] L. Breiman, "Random forests," Machine learning, vol. 45, no. 1, 2001, pp. 5–32. https://doi.org/10.1023/A:1010933404324.
[7] Biau, G., Scornet, E. A random forest guided tour. TEST 25, 197–227 (2016). https://doi.org/10.1007/s11749-016-0481-7.
[8] Chen, T., & Guestrin, C. (2016). XGBoost: A scalable tree boosting system. In R. B. L. et al. (Eds.), Proceedings of the 22nd ACM SIGKDD International Conference on Knowledge Discovery and Data Mining (pp. 785–794). https://doi.org/10.1145/2939672.2939785.
[9] Alao, J. O., & Karami, A. (2025). Harnessing social media sentiment for predictive insights into the 2023 Nigerian presidential election. IEEE Access, 13, 12345–12358.
[10] A. Liaw and M. Wiener, "Classification and regression by random forest," R News , vol. 2, no. 3, 2002, pp. 18- 22.
[11] S. Xia, G. Wang, Z. Chen, Y. Duan, and Q. Jiu, "Complete random forest based class noise filtering learning for improving the generalizability of classifiers," IEEE Trans. Know/. Data Eng., vol. 31, no. 11, November 2019, pp. 2063- 2078. https://doi.org/10.1109/TKDE.2018.2873791.
[12] M. A. Hannan, J. A. Ali, A. Mohamed, and M. N. Uddin, "A ran- dom forest regression based space vector PWM inverter controller for the induction motor d1ive," IEEE Trans. Ind. Electron., vol. 64, no. 4, April 2017, pp. 2689- 2699. https://doi.org/10.1109/TIE.2016.2631121.
[13] Z. Ge, Z. Song, S. X. Ding, and B. Huang, "Data mining and analytics in the process industry: The role of machine learning," IEEE Access, vol. 5, 2017, pp. 20590- 20616. https://doi.org/10.1109/ACCESS.2017.2756872.
[14] Lewis-Beck, M.S., Rice, T.W., Forecasting presidential elections: a comparison of naive models. Polit Behav, vol. 6, no. 1, 1984, pp. 9–21, 1984. https://doi.org/10.1007/BF00988226.

[15] Dwi Prasetyo N, Hauf C, Twitter-based election prediction in the developing world. In: Proceedings of the 26th ACM Conference on Hypertext & Social Media, ACM, 2015, pp 149–158. https://doi.org/10.1145/2700171.2791033.
[16] Kagan V, Stevens A, Subrahmanian V Using twitter sentiment to forecast the 2013 pakistani election and the 2014 indian election. IEEE Intell Syst 1:2–5, 2015. https://doi.org/10.1109/MIS.2015.16.
[17] Andreas Graefe AC (2014) State-by-state political economy model. https://pollyvote.com/en/components/econometric-models/jerom e-jerome/
[18] Campbell JE, Norpoth H et al, A recap of the 2016 election forecasts. PS: Polit Sci Polit 50(2):331–338, 2017. https://doi.org/10.1017/S1049096516002766.
[19] Tien C, Lewis-Beck MS, In forecasting the 2016 election result, modelers had a good year. pollsters did not. USApp–American Politics and Policy Blog, 2016.
[20] Dassonneville R, Lewis-Beck MS, Comparative election forecasting. synthetic models for europe. In: Conference on methodological innovations in the study of elections in Europe and beyond, College Station, 2014.
[21] Lewis-Beck MS, Tien CP, House forecasts: structure-x models for 2018. PS: Polit Sci Polit 51(S1):17–20. https://doi.org/10.1017/S1049096518001257.
[22] Dassonneville R, Lewis-Beck MS, Mongrain P, Forecasting Dutch elections: an initial model from the March 2017 legislative contests. Res Polit 4(3):2053168017720023, 2017. https://doi.org/10.1177/2053168017720023.
[23] Holbrook TM, Incumbency, national conditions, and the 2012 presidential election. PS: Polit Sci Polit 45(4):640–643, 2012. https://doi.org/10.1017/S1049096512000923.
[24] Traugott MW, Public opinion polls and election forecasting. PS: Polit Sci Polit 47(2):342–344, 2014. https://doi.org/10.1017/S1049096514000171.
[25] Blumenthal M, Polls, forecasts, and aggregators. PS: Polit Sci Polit 47(2):297–300, 2014. https://doi.org/10.1017/S1049096514000055.
[26] Silver N (2018) How fvethirtyeights house, senate and governor models work. https://fvethirtyeight.com/methodology/how-fvethirty eights-house-and-senate-models-work/. Accessed 08 October 2021.
[27] Feldman R, Techniques and applications for sentiment analysis. Communication ACM 56(4):82–89, 2013. https://doi.org/10.1145/2436256.2436274.
[28] Tumasjan A, Sprenger TO, Sandner PG, Welpe IM, Predicting elections with twitter: what 140 characters reveal about political sentiment. ICWSM 10(1):178–185, 2010. https://doi.org/10.1609/icwsm.v4i1.14009.
[29] Mislove A, Lehmann S, Ahn Y-Y, Onnela J-P, Rosenquist JN, Understanding the demographics of twitter users. ICWSM 11(5th):25, 2011.
[30] Mustafaraj E, Finn S, Whitlock C, Metaxas PT, Vocal minority versus silent majority: Discovering the opinions of the long tail. In: Privacy, Security, Risk and Trust (PASSAT) and 2011 IEEE Third Inernational Conference on Social Computing (SocialCom), 2011 IEEE Third International Conference on, IEEE, 2011, pp 103–110. https://doi.org/10.1109/PASSAT/SocialCom.2011.188.
[31] Skoric M, Poor N, Achananuparp P, Lim E-P, Jiang J , Tweets and votes: A study of the 2011 Singapore general election. In: System Science (HICSS), 2012 45th Hawaii International Conference on, IEEE, 2012, pp 2583–2591. https://doi.org/10.1109/HICSS.2012.607.
[32] Wan V, Campbell W. Support vector machines for speaker verifcation and identifcation. In: IEEE proceeding. 2000
[33] Yeung KY, Bumgarner RE, Raftery AE. Bayesian model averaging: development of an improved multi-class, gene selection and classifcation tool for microarray data. Bioinformatics. 2005;21:2394–402. https://doi.org/10.1093/bioinformatics/bti319.
[34] Quinlan JR, Rivest RL. Inferring decision trees using the minimum description length principle. Inform Comput. 1989;80(3):227–48. https://doi.org/10.1016/0890-5401(89)90010-2.