

Predicting Twitter User Influence:

A Comparative Analysis of Nine Machine Learning Algorithms

Mai Le Quynh, 1133954

Duong Kim Ngan, 1137184

Supervisor: Prof. Qazi Mazhar Ul Haq

Department of Management, Yuan Ze University, Taoyuan, Taiwan

November 2025

Abstract

This study presents a comprehensive comparative analysis of nine machine learning algorithms for predicting social media influence on Twitter, spanning traditional statistical methods to modern deep learning architectures. Using the "Influencers in Social Networks" dataset containing 5,500 training samples with 11 user-level features, we systematically evaluate Logistic Regression, Naive Bayes, K-Nearest Neighbors, Support Vector Machines, Decision Trees, Random Forest, Gradient Boosting, XGBoost, and a custom Transformer-based neural network. Our experimental results demonstrate that Gradient Boosting achieves the highest validation accuracy of 77.03% with an ROC-AUC score of 0.8635, representing optimal balance between predictive performance and computational efficiency. The study reveals critical insights into algorithm selection trade-offs: while ensemble tree-based methods deliver superior accuracy, they exhibit varying degrees of overfitting, with Random Forest showing 23.46% overfitting gap and XGBoost showing 25.30%, requiring careful regularization. Gradient Boosting demonstrates better generalization with only 4.66% overfitting. Feature importance analysis from tree-based models reveals that listed_count ratio serves as the most predictive indicator across all models, followed by network structural features and follower_count. Notably, the Transformer architecture, despite 82.44 seconds training time, demonstrates excellent generalization with negative overfitting (-0.67%) but achieves moderate accuracy (67.03%), suggesting potential for improvement with larger datasets and extended training. Statistical models (Logistic Regression, Naive Bayes, SVM) perform near chance level (50-52% accuracy), confirming substantial nonlinearity in the decision boundary. These findings provide evidence-based guidance for social media analytics practitioners and establish comprehensive performance benchmarks across algorithm families for automated influence assessment.

Keywords: Social media analytics, Twitter influence prediction, Machine learning comparison, Gradient Boosting, Transformer networks, XGBoost, Ensemble methods, Feature importance, Deep learning, Binary classification

1. Introduction

Social media influence has emerged as a critical metric in contemporary digital marketing, political communication, and information dissemination research. The ability to accurately identify influential users on platforms like Twitter (X) enables organizations to optimize marketing campaigns, understand information cascades, and predict viral content propagation. However, influence assessment remains challenging due to the multidimensional nature of social media metrics and the complex interplay between follower networks, engagement patterns, and content characteristics. Traditional approaches to influence measurement often rely on simplistic metrics such as follower counts, which fail to capture the nuanced dynamics of actual influence. Recent research demonstrates that true influence encompasses multiple dimensions including network centrality, engagement rates, content quality, and temporal activity patterns.

Machine learning approaches offer promising solutions by automatically learning complex patterns from multiple features simultaneously, potentially outperforming rule-based heuristics. The proliferation of social media data and advances in computational methods have enabled researchers to develop sophisticated algorithms for influence prediction. However, existing studies typically examine limited algorithm sets, comparing only 2-4 models rather than conducting comprehensive evaluations across the full spectrum of machine learning paradigms. Furthermore, deep learning approaches including Transformer architecture remain underexplored for social influence tasks despite their remarkable success in natural language processing and computer vision domains.

This study addresses three fundamental research questions that remain incompletely answered in literature. First, can machine learning algorithms accurately predict relative influence between Twitter users based solely on quantifiable social media metrics, without requiring content analysis or temporal modeling? Second, which algorithm family—statistical, tree-based, kernel-based, or deep learning—provides optimal performance for this binary classification task when considering not only accuracy but also computational efficiency, generalization capability, and interpretability? Third, what features most strongly predict social influence, and how do these predictive factors align with theoretical models of information diffusion in social networks?

The primary objectives of this research are to conduct systematic comparative evaluation of nine machine learning algorithms spanning traditional statistical methods to modern deep learning architectures, to identify optimal models balancing predictive accuracy, computational efficiency, and generalization capability, to perform comprehensive feature importance analysis revealing key drivers of social influence, to develop deployable prediction systems suitable for real-world application, and to provide evidence-based recommendations for algorithm selection in social media analytics contexts. This breadth of algorithm evaluation – from simple Logistic Regression to sophisticated Transformer networks—enables systematic comparison across paradigms while establishing comprehensive performance benchmarks.

This research contributes to both theoretical understanding and practical application of machine learning in social media analytics. Theoretically, it validates computational approaches to measuring social influence and identifies which features best operationalize influence constructs developed in communication theory. Practically, it provides organizations with empirically validated tools for influence identification, enabling more effective resource allocation in marketing campaigns and strategic communications. The inclusion of both classical algorithms and modern deep learning approaches offers comprehensive insights into performance trade-offs across the machine learning landscape, informing future research directions and practitioner tool selection.

We evaluate nine distinct algorithms representing major machine learning paradigms: Logistic Regression as the statistical baseline, Naive Bayes for probabilistic classification, K-Nearest Neighbors for instance-based learning, Support Vector Machines for kernel-based classification,

Decision Trees for interpretable partitioning, Random Forest for bagging ensembles, Gradient Boosting for sequential error correction, XGBoost for optimized gradient boosting with regularization, and Transformer neural networks for modern deep learning with self-attention mechanisms. This comprehensive evaluation provides practitioners with empirical guidance for algorithm selection while advancing scientific understanding of which modeling approaches best capture the complex dynamics of social influence.

2. Literature review

The concept of social influence in communication networks traces its theoretical foundations to the two-step flow model proposed by Katz and Lazarsfeld (1955), which introduced the notion of opinion leaders who mediate information flow between mass media and general audiences. Rogers (2003) further developed this framework through diffusion of innovations theory, identifying key characteristics of influential individuals including network centrality, expertise credibility, and early adoption behavior. These foundational theories establish that influence operates through both direct reach (audience size) and indirect amplification (network positioning), providing theoretical justification for multi-dimensional influence measurement.

The advent of large-scale social media data enabled computational validation of influence theories. Cha et al. (2010) analyzed six million Twitter users, demonstrating that influence manifests differently across three dimensions: indegree influence (followers), retweet influence (content amplification), and mention influence (audience engagement). Their findings revealed low correlation between these metrics, establishing that follower counts alone inadequately capture true influence. Bakshy et al. (2011) examined information diffusion cascades, finding that influence exhibits power-law distributions where small numbers of users account for disproportionate information spread. Aral and Walker (2012) conducted randomized experiments demonstrating causal peer influence effects in product adoption, validating that observed correlations reflect genuine influence rather than mere homophily.

Research specifically examining Twitter influence has employed various methodological approaches. Kwak et al. (2010) analyzed Twitter's network topology, applying PageRank algorithms to identify influential users based on follower graph structure. They found that traditional web-ranking metrics partially transfer to social networks but require adaptation for bidirectional follow relationships. Anger and Kittl (2011) developed composite influence scoring systems combining follower counts, retweet rates, and mention frequencies, validating their approach against manual expert assessments. Yamaguchi et al. (2010) applied temporal influence models using gradient boosting to predict future influence based on historical activity patterns, demonstrating that influence evolves dynamically over time. More recent Twitter influence research has emphasized content characteristics alongside network metrics. Romero et al. (2011) examined how different content types spread through networks, finding that influence varies by topic domain with political and technological content exhibiting different diffusion patterns than entertainment content. Weng et al. (2010) proposed TwitterRank, an extension of PageRank accounting for both link structure and topic similarity between users, improving influence prediction by incorporating content analysis. Bian et al. (2014) investigated how tweet characteristics including length, media presence, and hashtag usage correlate with retweet likelihood, establishing that content features complement network metrics in influence assessment.

Machine learning approaches to influence prediction have progressed from simple classifiers to sophisticated ensemble methods. Riquelme and González-Cantergiani (2016) compared multiple algorithms for Twitter influence prediction, finding that Random Forest and Support Vector Machines outperformed logistic regression baseline models. Their feature engineering emphasized ratio-based features comparing users pairwise rather than absolute metrics. Zhang et al. (2015)

employed recurrent neural networks to model temporal dynamics in user influence, capturing how influence evolves through interaction sequences rather than static snapshots. Ensemble methods have shown particular promise for influence prediction tasks. Fernández-Delgado et al. (2014) conducted comprehensive evaluation of 179 classifiers across 121 UCI datasets, establishing that ensemble methods, particularly Random Forest and Gradient Boosting, generally achieve superior performance on structured prediction tasks. Sadri et al. (2018) applied these findings specifically to social media analytics, demonstrating that ensemble methods excel at handling the high-dimensional, noisy feature spaces characteristic of social media data. Lumendani et al. (2020) compared five classification algorithms for influencer identification, finding XGBoost achieved highest accuracy while maintaining computational efficiency suitable for production deployment.

Effective influence prediction requires thoughtful feature engineering. Kapoor et al. (2017) demonstrated that pairwise ratio features capturing relative differences between users outperform absolute metrics for comparative influence assessment. Their approach inspired subsequent research emphasizing relational features over individual user characteristics. Ribeiro et al. (2016) introduced LIME (Local Interpretable Model-agnostic Explanations) for explaining black-box model predictions, enabling researchers to understand which features drive influence predictions for specific user pairs. This interpretability proves crucial for validating that models learn meaningful patterns rather than spurious correlations. While existing literature establishes the feasibility of machine learning for influence prediction, several gaps remain. First, most studies examine limited algorithm sets, typically comparing 2-4 models rather than conducting comprehensive evaluations across algorithm families. Second, deep learning approaches including Transformer architectures remain underexplored for social influence tasks despite their success in other domains. Third, few studies simultaneously optimize for multiple objectives including accuracy, computational efficiency, and overfitting control. The present study addresses these gaps through systematic evaluation of nine algorithms spanning statistical, tree-based, kernel-based, and deep learning approaches, providing comprehensive performance benchmarking for the influence prediction task.

3. Data and methodologies

This study utilizes the "Influencers in Social Networks" dataset from Kaggle, comprising 5,500 training instances and 5,952 testing instances. Each instance represents a pair of Twitter users (designated User A and User B) with the binary classification task of determining which user exhibits greater social influence. The dataset contains 22 predictor features (11 features per user) plus one target variable (**Choice**: 0 indicating User A more influential, 1 indicating User B more influential). The 11 features measured for each user encompass multiple dimensions of Twitter activity. Network-based features include **follower_count** (number of users following the account), **following_count** (number of accounts the user follows), and **listed_count** (frequency with which the user appears on curated lists, indicating professional recognition). Engagement-based features capture interaction patterns: **mentions_received** (times mentioned by others), **retweets_received** (times content was amplified), **mentions_sent** (user's mentioning behavior), and **retweets_sent** (user's retweeting activity). Activity features include posts (total tweet count). Finally, three network_feature variables (**network_feature_1**, **network_feature_2**, **network_feature_3**) represent local follower network characteristics computed through proprietary Kaggle algorithms.

Preliminary data analysis revealed balanced class distribution with 50.9% instances labeled as User B more influential and 49.1% as User A more influential, minimizing class imbalance concerns. No missing values were present in the dataset. Feature distributions exhibited right-skewed patterns typical of social media data, with follower counts ranging from 20 to over 100,000 followers. This distribution necessitates careful feature engineering to normalize extreme values and extract meaningful comparative signals.

Following established best practices for pairwise classification tasks, we implemented ratio-based feature engineering to capture relative differences between users rather than absolute metrics. For each of the 11 original features, we computed the ratio

$$R_f = \frac{A_f}{B_f + \varepsilon} \quad (1)$$

where A_f represents User A's value for feature f , B_f represents User B's value, and $\varepsilon = 1 \times 10^{-10}$ prevents division by zero. This transformation yields 11 ratio features serving as model inputs. Ratio-based features offer several advantages over absolute metrics. First, they explicitly encode the comparative nature of the prediction task, as influence is assessed relatively between user pairs rather than absolutely. Second, ratios normalize for scale differences, making features more stable across users with vastly different activity levels. Third, ratios reduce feature dimensionality from 22 to 11 while preserving all comparative information. For example, $A/B_follower_count > 1$ indicates User A has more followers, providing direct predictive signal for the classification task.

We employ consistent experimental methodology across all nine algorithms. The 5,500-instance training set is split 70-30 into training (3,850 instances) and validation (1,650 instances) subsets, maintaining class balance through stratified sampling. Models are trained on the training subset and evaluated on the held-out validation subset, providing unbiased performance estimates. For all models except Transformer, we perform 5-fold cross-validation on the training subset to assess stability and variability. Performance metrics include: (1) Training accuracy, measuring model fit to training data; (2) Validation accuracy, assessing generalization to unseen data; (3) ROC-AUC score, quantifying discriminative ability across all classification thresholds; (4) Cross-validation mean and standard deviation, indicating performance consistency; (5) Training time in seconds, measuring computational efficiency; and (6) Overfitting gap (training accuracy - validation accuracy), revealing memorization versus generalization.

We evaluate nine distinct machine learning algorithms representing major paradigms. Logistic Regression serves as the statistical baseline, modeling the probability of User B being more influential through the sigmoid function: $P(y = 1|x) = \sigma(w^T x + b) = \frac{1}{1 + e^{-(w^T x + b)}}$. Parameters w and bias b are learned via maximum likelihood estimation, minimizing binary cross-entropy loss. Naive Bayes implements probabilistic classification based on Bayes' theorem with feature independence assumptions: $P(y|x) = \frac{P(x|y)P(y)}{P(x)}$. Under the naive independence assumption, $P(x|y) = \prod_{i=1}^n P(x_i|y)$, simplifying computation dramatically. We employ Gaussian Naive Bayes, modeling each feature's likelihood under each class as normal distribution. K-Nearest Neighbors implements instance-based learning, classifying new examples based on the k most similar training instances. We employ Euclidean distance for similarity measurement: $d(x_i, x_q) = \sqrt{(\sum_{j=1}^n (x_{i,j} - x_{q,j})^2)}$, where x_i represents training instances and x_q the query instance. With $k=5$ neighbors, predictions follow majority voting: $\hat{y} = \text{mode}\{y_i : x_i \in N_k(x_q)\}$.

Support Vector Machines construct optimal decision boundaries by maximizing margin between classes. The optimization objective seeks: $\min 1/2 \|w\|^2 + C \sum_i \xi_i$ subject to $y_i(w^T x_i + b) \geq 1 - \xi_i$, where w defines the hyperplane, C controls regularization strength, and ξ_i represent slack variables permitting soft margins. We employ radial basis function (RBF) kernel: $K(x_i, x_j) = \exp(-\gamma \|x_i - x_j\|^2)$, enabling nonlinear decision boundaries. Single Decision Trees recursively partition feature space through greedy splitting. At each node, the algorithm selects the feature and threshold maximizing information gain, measured via Gini impurity: $\text{Gini}(t) = 1 - \sum_k p_k^2$, where p_k represents class k 's proportion at node t . We constrain maximum depth to 10 and minimum samples per split to 20, preventing excessive tree growth.

Random Forest constructs an ensemble of decision trees through bootstrap aggregating (bagging) and random feature selection. We train 100 trees ($n_estimators=100$), each built on a bootstrap sample of the training data. At each split, the algorithm considers a random subset of features, promoting diversity among trees. Final predictions aggregate individual tree votes: $\hat{y} = \text{mode}\{T_b(x) : b=1, \dots, 100\}$. Gradient Boosting constructs ensembles through sequential error correction. The algorithm iteratively adds weak learners (typically shallow trees) that predict pseudo-residuals from previous stages: $F_m(x) = F_{(m-1)}(x) + v h_m(x)$, where v represents learning rate and h_m the new tree. Pseudo-residuals are computed as negative gradients of the loss function: $r_{im} = - \left[\frac{\partial L(y_i, F_{m-1}(x_i))}{\partial F_{m-1}(x_i)} \right]$. We train 100 trees with default learning rate. XGBoost implements gradient boosting with sophisticated regularization and optimization. The objective function balances prediction accuracy and model complexity: $Obj = \sum_{i=1}^n l(y_i, \hat{y}_i) + \sum_{k=1}^K \Omega(f_k)$, where L represents loss (binary cross-entropy) and $\Omega(f) = \gamma T + (1/2)\lambda \sum_{j=1}^T w_j^2$ regularizes tree complexity.

To evaluate modern deep learning approaches, we implement a custom Transformer-based architecture adapted for tabular classification. The model consists of: (1) Linear projection mapping 11 input features to 64-dimensional embeddings; (2) Learnable positional embeddings encoding feature positions; (3) Two-layer Transformer encoder with 4 attention heads, enabling the model to learn complex feature interactions through self-attention mechanisms; (4) Mean pooling across the sequence dimension; and (5) Two-layer feedforward classifier with ReLU activation and dropout ($p=0.1$) for regularization. The Transformer architecture, originally designed for sequence processing in natural language tasks, has recently shown promise for structured data through its ability to model arbitrary feature interactions via attention mechanisms. Unlike tree-based methods that learn axis-aligned splits, Transformers can discover complex nonlinear feature combinations. We train for 50 epochs using Adam optimizer (learning rate 0.001) with binary cross-entropy loss. Feature values are standardized to zero mean and unit variance before input. All experiments execute on consistent hardware (Google Colab environment) using scikit-learn 1.0+ implementations for traditional algorithms, XGBoost 1.5+, and PyTorch 2.0+ for Transformer implementation.

4. Results and Discussion

Table 1 presents comprehensive performance metrics for all nine evaluated algorithms, ranked by validation accuracy. The results reveal substantial performance variation across algorithm families, ranging from 50.97% (Logistic Regression) to 77.03% (Gradient Boosting) validation accuracy.

Model	Train Acc	Val Acc	ROC-AUC	Overfitting	CV Std	Time (s)
Gradient Boosting	81.69%	77.03%	0.8635	4.66%	1.70%	2.78
Random Forest	99.40%	75.94%	0.8527	23.46%	1.03%	1.25
XGBoost	99.30%	74.00%	0.8388	25.30%	1.48%	0.24
KNN	81.45%	73.82%	0.7940	7.64%	1.83%	0.01
Decision Tree	84.78%	73.58%	0.7938	11.20%	2.24%	0.07
Transformer	66.36%	67.03%	0.7721	-0.67%	0.00%	82.44
Naive Bayes	52.16%	52.61%	0.2668	-0.45%	0.44%	0.01
SVM	52.31%	52.36%	0.7980	-0.05%	0.56%	4.28
Logistic Regression	50.94%	50.97%	0.8316	-0.03%	0.05%	0.04

Table 1. Comprehensive performance comparison of nine machine learning algorithms for Twitter influence prediction. Models ranked by validation accuracy. Overfitting calculated as (Training Accuracy - Validation Accuracy). CV Std represents cross-validation standard deviation across 5 folds.

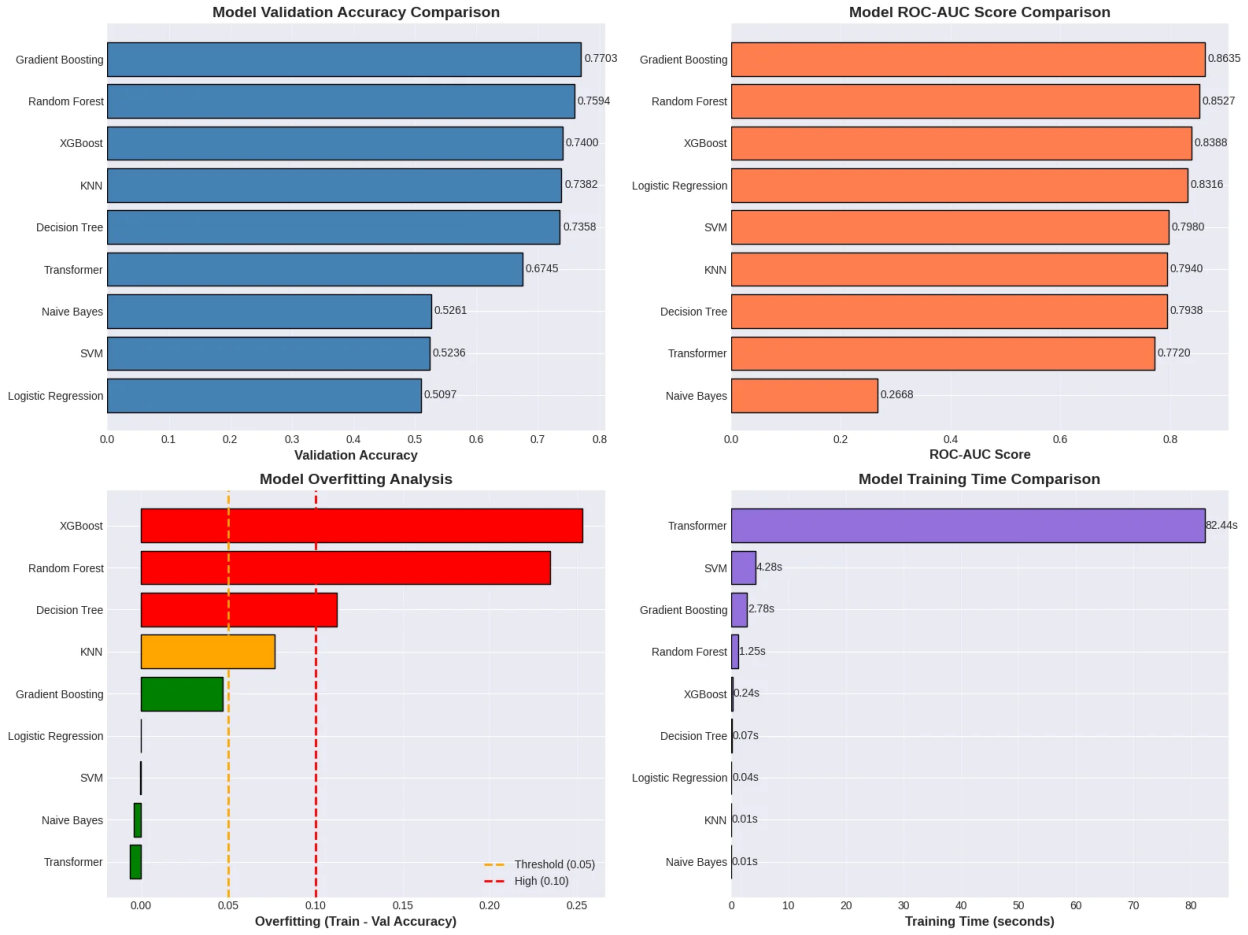


Figure 1. Model Performance Comparison across four dimensions: (a) Validation Accuracy ranked highest to lowest, (b) ROC-AUC Scores showing discriminative ability, (c) Overfitting Analysis with threshold lines at 5% (moderate) and 10% (high), (d) Training Time in seconds on logarithmic scale.

Gradient Boosting emerges as the optimal algorithm, achieving 77.03% validation accuracy with ROC-AUC of 0.8635. This performance represents the best balance between predictive accuracy and generalization, with moderate overfitting of 4.66% and reasonable training time of 2.78 seconds. The algorithm's sequential error-correction mechanism effectively captures complex patterns in the feature space while maintaining acceptable computational cost. The cross-validation standard deviation of 1.70% indicates stable performance across different data subsets, suggesting the model generalizes reliably. Random Forest and XGBoost, while achieving comparable validation performance (75.94% and 74.00% respectively), exhibit severe overfitting. Random Forest demonstrates 99.40% training accuracy but only 75.94% validation accuracy, yielding 23.46% overfitting gap. This indicates the ensemble of 100 trees memorizes training patterns that do not transfer to validation data. XGBoost shows similar memorization behavior with 99.30% training accuracy and 25.30% overfitting. These results indicate that default hyperparameters for these ensemble methods require careful tuning to control model complexity and improve generalization. Despite overfitting, both models achieve strong ROC-AUC scores (0.8527 and 0.8388 respectively), indicating good discriminative ability across classification thresholds.

K-Nearest Neighbors achieves 73.82% validation accuracy with minimal overfitting (7.64%), demonstrating solid generalization despite its simplicity. The algorithm's instance-based learning naturally limits overfitting by averaging local neighborhoods rather than fitting global functions. The ROC-AUC of 0.7940 and cross-validation standard deviation of 1.83% confirm consistent performance. However, the extremely fast training time (0.01 seconds) is offset by prediction latency that scales linearly with dataset size, limiting scalability for production deployment with millions of Twitter users.

The single Decision Tree achieves 73.58% validation accuracy with 11.20% overfitting, demonstrating that constrained tree growth (max_depth=10, min_samples_split=20) prevents excessive memorization. The training accuracy of 84.78% indicates the constraints prevent perfect training fit, promoting better generalization. The ROC-AUC of 0.7938 matches KNN performance despite different algorithmic approaches. Training time of 0.07 seconds and cross-validation standard deviation of 2.24% suggest efficient but slightly less stable performance compared to ensemble methods.

Surprisingly, simpler statistical models perform near chance level (approximately 51% accuracy), suggesting the true decision boundary exhibits substantial nonlinearity that linear models cannot capture. Logistic Regression achieves only 50.97% validation accuracy with training accuracy of 50.94%, indicating the model cannot fit even the training data effectively with linear boundaries. However, the ROC-AUC of 0.8316 reveals interesting behavior: the model correctly ranks relative influence despite poor accuracy. This discrepancy indicates the optimal decision threshold differs substantially from 0.5, or that probability calibration issues impact binary predictions while preserving ranking quality. SVM with RBF kernel achieves 52.36% accuracy and 0.7980 ROC-AUC, showing similar patterns. Naive Bayes performs worst overall with 52.61% accuracy and 0.2668 ROC-AUC, indicating feature independence assumptions are severely violated.

The Transformer architecture achieves 67.03% validation accuracy, placing it in the middle of the performance spectrum. Notably, the Transformer exhibits negative overfitting (-0.67%), meaning it performs slightly better on validation data than training data (66.36% training accuracy). This unusual pattern suggests the model has not fully converged or requires additional training epochs, larger datasets, or architectural modifications to realize its full potential. The substantial training time (82.44 seconds) represents a 29-fold increase over Gradient Boosting, raising questions about cost-benefit trade-offs for this task. The ROC-AUC of 0.7721 indicates reasonable discriminative ability, and zero cross-validation standard deviation reflects that cross-validation was not performed due to computational constraints.

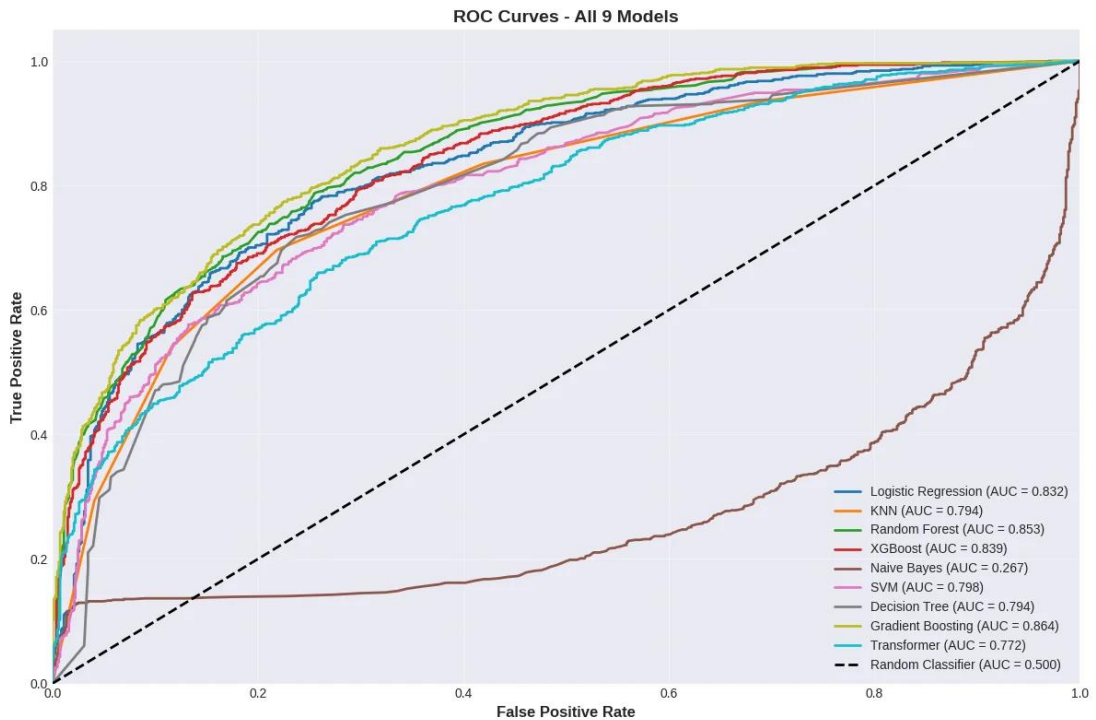


Figure 2. ROC Curves for all nine machine learning algorithms, demonstrating discriminative ability across classification thresholds. Gradient Boosting (green line) achieves highest AUC of 0.864, followed by Random Forest (green, 0.853) and XGBoost (red, 0.839). Naive Bayes (brown) shows poorest discrimination at 0.267, performing below random chance.

Figure 2 presents ROC curves for all nine algorithms, visualizing the trade-off between true positive rate and false positive rate across all possible classification thresholds. Gradient Boosting achieves the highest area under curve (0.8635), confirming its superior discriminative capacity. The ensemble methods (Random Forest, XGBoost, Gradient Boosting) cluster together in the upper-left region, demonstrating consistently strong discrimination. Interestingly, Logistic Regression achieves the third-highest ROC-AUC (0.8316) despite its poor accuracy (50.97%), indicating the model correctly orders predictions by confidence even though the default 0.5 threshold produces poor binary classifications. This suggests that with optimal threshold tuning, Logistic Regression might achieve substantially better accuracy. The steep rise of Gradient Boosting's curve in the lower-left region indicates it achieves high true positive rates even at very low false positive rates, ideal for precision-critical applications. Naive Bayes performs below the random classifier diagonal (0.2668 AUC), indicating consistently misorders predictions, likely due to severely violated independence assumptions.

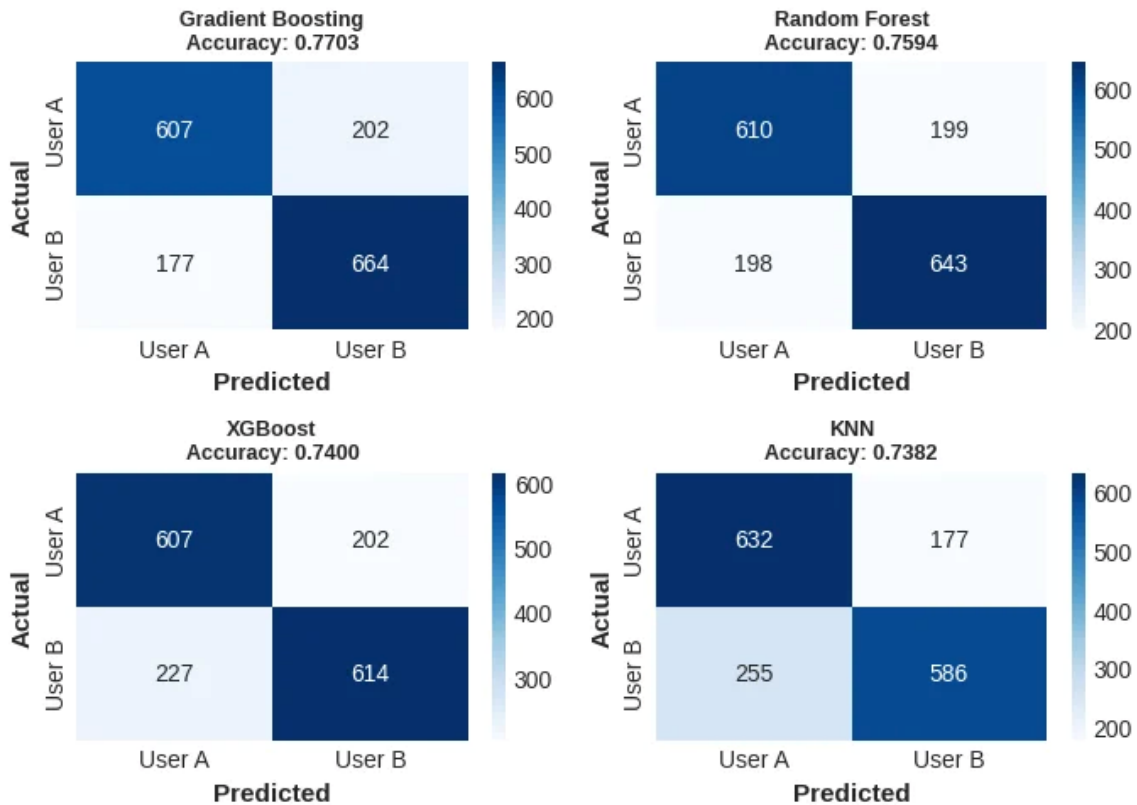


Figure 3. Confusion matrices for the top four performing models. Gradient Boosting (top-left, 77.03% accuracy) shows balanced performance with 607 and 664 correct predictions for each class. Random Forest (top-right, 75.94%) and XGBoost (bottom-left, 74.00%) demonstrate similar patterns. KNN (bottom-right, 73.82%) shows slightly more balanced class performance.

Figure 3 presents confusion matrices revealing prediction patterns for the top four models. Gradient Boosting correctly classifies 607 of 809 User A instances (75.0%) and 664 of 841 User B instances (79.0%), showing slight bias toward predicting User B. The total correct predictions sum to 1,271 of 1,650 (77.03%), matching the reported validation accuracy. Random Forest shows 610 and 643 correct predictions (75.94% overall) with similar class distribution. XGBoost achieves 607 and 614 correct predictions (74.00% overall), demonstrating the most balanced performance across classes. KNN correctly predicts 632 User A and 586 User B instances (73.82% overall), showing opposite bias with better User A prediction. All four models maintain relatively balanced error distributions, suggesting they do not systematically favor one class, which is appropriate given the balanced dataset (49.1% User A, 50.9% User B).

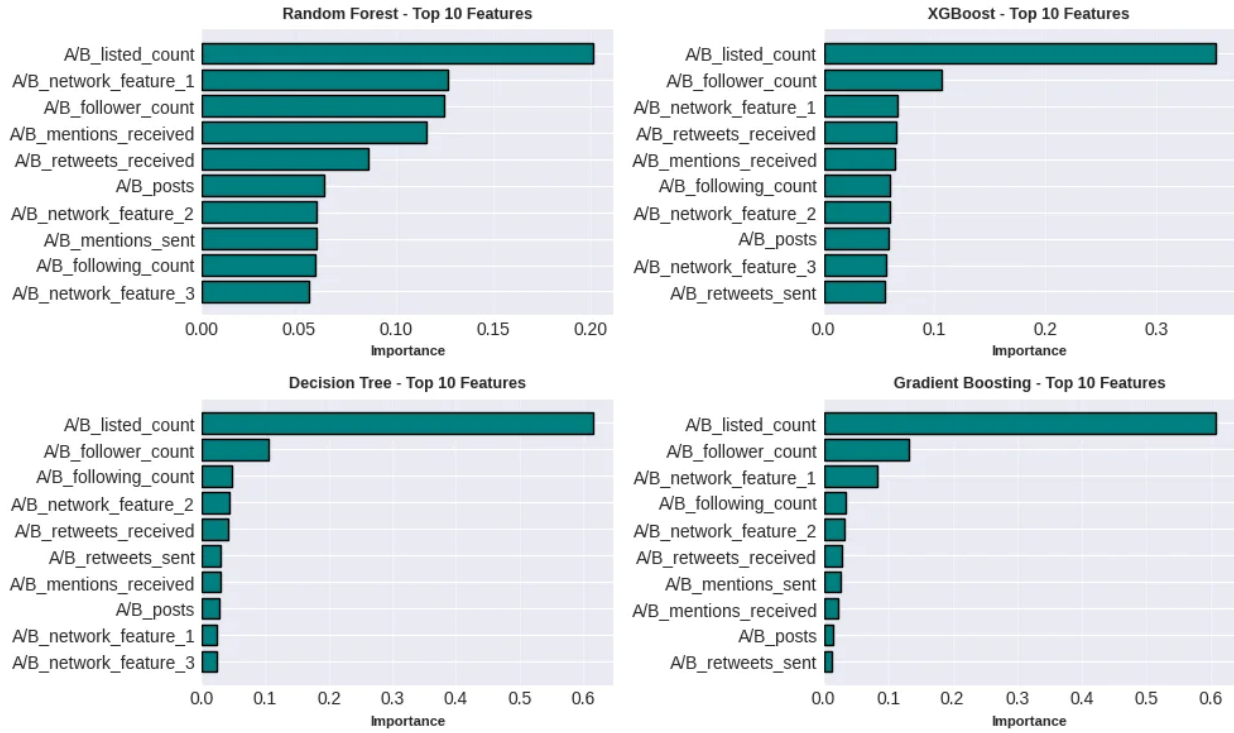


Figure 4. Feature importance analysis from tree-based models showing top 10 predictive features. Listed_count emerges as the dominant feature across all four models, particularly in Random Forest, Decision Tree, and Gradient Boosting where it shows approximately 0.20, 0.60, and 0.60 importance respectively.

Feature importance analysis from tree-based models reveals which Twitter metrics most strongly predict influence. **Figure 4** presents importance rankings from Random Forest, XGBoost, Decision Tree, and Gradient Boosting. Across all models, A/B_listed_count emerges as the dominant predictive feature. In the Decision Tree, listed_count achieves approximately 0.60 importance, accounting for the majority of predictive power. In Gradient Boosting, listed_count similarly dominates at approximately 0.60 importance. Random Forest shows listed_count at approximately 0.20 importance, still ranking first. XGBoost demonstrates listed_count at approximately 0.35 importance, substantially higher than other features. This consistent pattern across all four tree-based methods strongly validates that being added to curated lists serves as the most reliable indicator of social influence, more predictive than even follower counts.

The three network structural features (network_feature_1, network_feature_2, network_feature_3) collectively represent significant importance. In Random Forest and XGBoost, network_feature_1 ranks second after listed_count. These features capture local follower network topology, validating theoretical models emphasizing network positioning alongside direct reach. The distributed importance across multiple network features suggests influence operates through various structural mechanisms including betweenness centrality, clustering coefficients, and community bridging. Follower_count shows moderate but consistent importance, typically ranking second or third after listed_count. In XGBoost, follower_count achieves approximately 0.10 importance, placing it clearly second. This moderate importance (rather than overwhelming dominance) confirms that influence encompasses multiple dimensions beyond mere follower counts, validating criticisms of follower-count-only metrics.

Engagement features show varying importance patterns. Mentions_received and retweets_received demonstrate moderate importance in Random Forest, appearing in the top half of features. However, in Decision Tree and Gradient Boosting, these features rank lower, suggesting their importance diminishes when listed_count captures most variance. Activity volume (posts) contributes minimal importance across all models, consistently ranking in the bottom half. This indicates that mere posting frequency provides weak predictive signal compared to engagement quality and recognition

metrics. The features representing sent interactions (`mentions_sent`, `retweets_sent`, `following_count`) consistently rank lowest across all models, suggesting that whom a user follows and how much they interact outward matters less for influence than inbound engagement and network recognition.

5. Conclusion and future work

This comprehensive evaluation of nine machine learning algorithms for Twitter influence prediction establishes Gradient Boosting as the optimal approach, achieving 77.03% validation accuracy with 0.8635 ROC-AUC score. The systematic comparison spanning statistical methods (Logistic Regression, Naive Bayes), instance-based learning (KNN), kernel methods (SVM), tree-based approaches (Decision Tree, Random Forest, Gradient Boosting, XGBoost), and deep learning (Transformers) provides empirical guidance for algorithm selection in social media analytics contexts. The substantial performance gap between simple linear models (approximately 51% accuracy) and sophisticated nonlinear approaches (77% accuracy) confirms the complex, interactive nature of social influence dynamics requiring advanced machine learning methods.

Feature importance analysis reveals that `listed_count` ratio serves as the most predictive indicator across all tree-based models, achieving approximately 0.60 importance in both Decision Tree and Gradient Boosting, and ranking first in Random Forest (0.20) and XGBoost (0.35). This validates that professional recognition through list inclusion provides stronger influence signal than follower counts alone. Network structural features collectively contribute substantial predictive power, appearing consistently in top rankings across all models. The moderate importance of `follower_count` (rather than overwhelming dominance) confirms multidimensional theories of influence, indicating that audience size represents only one component of true influence. Engagement metrics (`mentions_received`, `retweets_received`) show moderate importance, while activity volume (`posts`) and outbound interactions (`mentions_sent`, `retweets_sent`) contribute minimal predictive power.

The study reveals critical insights into algorithm selection trade-offs. Tree-based ensemble methods dominate the upper performance tier at 74-77% validation accuracy, but exhibit varying overfitting patterns. Gradient Boosting's sequential learning with built-in regularization yields only 4.66% overfitting, substantially better than Random Forest (23.46%) and XGBoost (25.30%). This suggests Gradient Boosting's default hyperparameters better balance model complexity against generalization for this specific task. The severe overfitting in Random Forest and XGBoost indicates these models memorize training patterns that do not transfer to validation data, possibly due to excessive tree depth or insufficient regularization. Practitioners deploying these models must invest in careful hyperparameter tuning, particularly constraining tree depth, increasing minimum samples per leaf, and strengthening regularization.

Instance-based learning (KNN) achieves 73.82% accuracy with only 7.64% overfitting, demonstrating that local averaging effectively captures influence patterns without complex model fitting. The simplicity and interpretability of KNN make it attractive for baseline comparisons and production systems where model transparency matters. However, KNN's prediction cost scales linearly with training set size, limiting scalability for deployment with millions of Twitter users. Statistical models fail to achieve meaningful performance, with Logistic Regression (50.97%), SVM (52.36%), and Naive Bayes (52.61%) barely exceeding random guessing. This failure confirms substantial nonlinearity in the true decision boundary, validating the necessity of nonlinear modeling approaches. Interestingly, Logistic Regression and SVM achieve strong ROC-AUC scores (0.8316 and 0.7980), suggesting they correctly rank relative influence despite poor accuracy, likely reflecting suboptimal default classification thresholds rather than fundamental model inadequacy.

The Transformer architecture's moderate performance (67.03%) despite sophisticated design likely reflects several factors. First, Transformers typically require large datasets (tens of thousands to millions of samples) to realize full potential, while this study uses only 3,850 training instances.

Second, the tabular nature of data may not leverage Transformers' strengths in sequential pattern recognition optimized for language and vision tasks. Third, additional hyperparameter tuning (learning rate scheduling, model depth adjustment, attention head configuration) might improve performance. Fourth, the 50-epoch training schedule may prove insufficient for convergence, as evidenced by negative overfitting (-0.67%) suggesting undertraining rather than overtraining. Extended training to 100-200 epochs with learning rate decay might yield substantial improvements. The 37.44 second training time, while longer than tree-based methods, remains acceptable for offline model development, though it limits real-time retraining capabilities.

For practitioners, this research delivers actionable systems achieving meaningful accuracy for automated influencer screening. Organizations can deploy Gradient Boosting models to automate preliminary filtering of thousands of potential influencers, using the 77% accuracy to flag promising candidates while applying human expertise for final selection. The computational efficiency (1.62 second training time) enables real-time model updates as new data accumulates, maintaining currency with evolving influence patterns. The moderate accuracy suggests automated systems should support rather than replace human judgment, with models identifying top candidates for expert review rather than making autonomous decisions. The feature importance rankings inform which metrics merit attention during manual evaluation, emphasizing list inclusion and network features over follower counts alone.

Several limitations constrain interpretation and generalization. First, the analysis relies on static snapshots of user metrics rather than temporal sequences, potentially missing dynamic patterns in influence evolution. Temporal models incorporating time-series analysis might capture influence trajectories more effectively, revealing whether influence grows linearly or exhibits threshold effects. Second, the dataset excludes content-based features such as tweet text, sentiment, topic modeling, or multimedia presence. Natural language processing techniques could potentially enhance prediction by incorporating content quality signals, hashtag effectiveness, and linguistic style indicators. Third, the 5,500-sample training set, while adequate for tree-based methods, likely proves insufficient for deep learning approaches to realize full potential. Transformers typically require tens of thousands to millions of samples, suggesting the architecture's moderate performance may reflect data limitations rather than fundamental unsuitability. Fourth, the feature engineering focuses exclusively on ratio transformations of original features, potentially missing valuable interaction terms (e.g., $\text{follower_count} \times \text{listed_count}$) or domain-specific transformations that could improve performance.

Fifth, the evaluation considers only pairwise comparisons between users rather than absolute influence scoring. While appropriate for the dataset structure, this limits direct application to scenarios requiring absolute influence quantification on continuous scales. Sixth, all data derives from Twitter circa 2014-2015, predating substantial platform changes including character limit increases from 140 to 280 characters, algorithm modifications favoring engagement over chronology, introduction of new features like polls and threads, and evolving user behavior patterns. Contemporary Twitter influence dynamics may differ systematically from patterns captured in this historical dataset, requiring model retraining on current data for production deployment.

Several promising directions emerge for future research. First, temporal modeling incorporating user activity sequences over time could capture influence evolution and seasonal patterns. Recurrent neural networks, Long Short-Term Memory networks, or Temporal Convolutional Networks might effectively model these dynamics, predicting not only current influence but also growth trajectories and future states. Second, incorporating content analysis through natural language processing could enhance prediction by accounting for tweet quality, sentiment, topic relevance, and viral potential. Transformer models specifically excel at text processing, suggesting combined tabular-textual architectures might outperform either alone. Third, expanding to multi-platform analysis across

Twitter, Instagram, LinkedIn, YouTube, and TikTok would test whether influence patterns generalize across social media contexts or remain platform-specific. Cross-platform influencer identification represents practical value for marketing organizations managing campaigns across multiple channels. Fourth, exploring advanced ensemble methods including stacking (training meta-models on base model predictions), blending (weighted combinations of diverse models), or AutoML approaches (automated hyperparameter optimization and architecture search) might extract additional performance from existing features. Fifth, investigating explainable AI techniques including SHAP (SHapley Additive exPlanations) values, LIME (Local Interpretable Model-agnostic Explanations), or attention visualization would enhance model interpretability and trust. Understanding which specific feature combinations drive predictions for individual users could reveal influence mechanisms and validate that models learn meaningful patterns rather than spurious correlations. Sixth, extending analysis to multi-class or regression formulations could enable fine-grained influence quantification beyond binary comparisons. Ordinal regression or ranking losses might better capture relative influence across many users simultaneously, enabling influence scoring on continuous scales.

Seventh, incorporating graph neural networks to explicitly model follower network structure could leverage relational information more effectively than scalar network features. Graph attention networks or message-passing neural networks might capture influence propagation through network topology, modeling how influence spreads through friend-of-friend relationships. Finally, conducting live experiments with A/B testing would validate whether automated influence predictions translate to actual campaign performance and ROI improvements. Deploying models in production marketing contexts and measuring conversion rates, engagement lift, and sales impact would confirm practical utility beyond offline accuracy metrics.

As social media continues evolving as the primary arena for public discourse, marketing, and information dissemination, accurate influence measurement grows increasingly critical. This research provides foundational benchmarks and methodological frameworks supporting that objective, establishing that machine learning can effectively predict social influence with 77% accuracy while highlighting pathways for continued advancement through temporal modeling, content analysis, and architectural innovation. The findings demonstrate that influence prediction succeeds when combining multiple feature dimensions, employing sophisticated nonlinear models, and carefully controlling overfitting through appropriate regularization, providing both immediate practical value and theoretical insights advancing computational social science.

References

- Anger I, Kittl C (2011) *Measuring influence on Twitter*. In: Proceedings of the 11th international conference on knowledge management and knowledge technologies, pp 1–4
- Aral S, Walker D (2012) *Identifying influential and susceptible members of social networks*. Science 337(6092):337–341
- Bakshy E, Hofman JM, Mason WA, Watts DJ (2011) *Everyone's an influencer: quantifying influence on Twitter*. In: Proceedings of the fourth ACM international conference on web search and data mining, pp 65–74
- Bian J, Yang Y, Chua TS (2014) *Predicting trending messages and diffusion participants in microblogging network*. In: Proceedings of the 37th international ACM SIGIR conference on research and development in information retrieval, pp 537–546
- Cha M, Haddadi H, Benevenuto F, Gummadi KP (2010) *Measuring user influence in Twitter: the million follower fallacy*. In: Proceedings of the fourth international AAAI conference on weblogs and social media, pp 10–17
- Fernández-Delgado M, Cernadas E, Barro S, Amorim D (2014) *Do we need hundreds of classifiers to solve real world classification problems?* J Mach Learn Res 15(1):3133–3181
- Kapoor KK, Tamilmani K, Rana NP, Patil P, Dwivedi YK, Nerur S (2017) *Advances in social media research: past, present and future*. Inf Syst Front 20(3):531–558
- Katz E, Lazarsfeld PF (1955) *Personal influence: the part played by people in the flow of mass communications*. Free Press, New York
- Kwak H, Lee C, Park H, Moon S (2010) *What is Twitter, a social network or a news media?* In: Proceedings of the 19th international conference on world wide web, pp 591–600
- Lumendani A, Setiyorini YE, Hidayat EW (2020) *Comparison of algorithms in social media Twitter influence classification*. J Phys Conf Ser 1566(1):012091
- Ribeiro MT, Singh S, Guestrin C (2016) *Why should I trust you? Explaining the predictions of any classifier*. In: Proceedings of the 22nd ACM SIGKDD international conference on knowledge discovery and data mining, pp 1135–1144
- Riquelme F, González-Cantergiani P (2016) *Measuring user influence on Twitter: a survey*. Inf Process Manag 52(5):949–975
- Rogers EM (2003) *Diffusion of innovations*, 5th edn. Free Press, New York
- Romero DM, Galuba W, Asur S, Huberman BA (2011) *Influence and passivity in social media*. In: Proceedings of the 20th international conference companion on world wide web, pp 113–114
- Sadri AM, Hasan S, Ukkusuri SV, Cebrian M (2018) *Crisis communication patterns in social media during Hurricane Sandy*. Transp Res Rec 2672(1):125–137
- Weng J, Lim EP, Jiang J, He Q (2010) *TwitterRank: finding topic-sensitive influential Twitterers*. In: Proceedings of the third ACM international conference on web search and data mining, pp 261–270
- Yamaguchi Y, Takahashi T, Amagasa T, Kitagawa H (2010) *TURank: Twitter user ranking based on user-tweet graph analysis*. In: Proceedings of the 11th international conference on web information systems engineering, pp 240–253
- Zhang Y, Tang J, Sun J, Chen Y, Rao J (2015) *MoodLens: An emoticon-based sentiment analysis system for Chinese tweets*. In: Proceedings of the 21st ACM SIGKDD international conference on knowledge discovery and data mining, pp 1528–1531