



Enhanced Machine Learning Framework for Detecting Fake News Used as Soft Power in Textual Content

Ahmed H Y Al-Noori¹, Moahaimen Talib², Ossama Ismail³ and Emad Tariq⁴.

¹Department of Artificial Intelligence and Robotics, College of Engineering, Al-Nahrain University,

² College of Engineering, Al-Bayan University

³Department of Biology, College of Science, Baghdad University, Baghdad, Iraq.

⁴Business Administration, College of Business Administration, Prince Mohammad bin Fahd University, Al-Khobar, Kingdom of Saudi Arabia.

• These authors contributed equally to this work

DOI: <https://www.doi.org/10.29194/NJES.29030521>

Received: Oct. 14, 2025

Revised: March 4, 2026

Accepted: March 29, 2026

Published: Sep 20, 2026

Abstract

The utilisation of fake news as a tool of soft power is regarded as a strategic component of contemporary warfare, posing an escalating threat to societal and political stability as well as public trust. Deep Learning models achieve superior performance; yet, they require substantial computational resources and possess interpretative restrictions, which hinder their practical application in real-time scenarios and impose constraints on resource configurations.

This study evaluates a traditional machine-learning framework based on linear and ensemble classifiers, including Linear Support Vector Classification (LinearSVC), Logistic Regression, Multinomial Naïve Bayes, and Random Forest. The framework employs a fixed, explicitly specified TF-IDF vectorization configuration and a consistent preprocessing pipeline. The system was assessed using the WELFake dataset, which contains over 70,000 labelled news stories, employing accuracy, precision, recall, F1-score, and ROC-AUC as evaluation metrics. The findings demonstrate the superiority of LinearSVC, attaining the maximum accuracy of 95.65%, providing balanced performance metrics, and surpassing all other models. This study contributes by providing an interpretable, scalable, and domain-agnostic solution through the development of a practical fake news detection system designed to combat digital disinformation in real-world contexts

Keywords: *Classification, AI, Machine Learning, Deep Learning, Fake News.*

Corresponding author: Provide the corresponding author information and publisher here. E-mail address: ahmed.hani.al-noori@ced.nahrainuniv.edu.iq

1. Introduction

The progress of fake news from a simple effect to a critical danger can change the societal aspects. This will lead to intrinsic changes in the general speech and the stability of the political and cultural systems [1]. In the past few decades, misinformation hasn't been used as random news but as soft power to impact people's beliefs and trust in the social and political realms. Sometimes this leads to unrest in these societies [2]. The hostile entities can easily use the digital platforms to tell false stories that can affect the social coherence and even deny the true evidence given by legal governments, sometimes leading to civil wars in countries [3, 4].

Traditional fact-checking methods struggle to manage the significant rise in web content generated and disseminated [5]. In contrast, automated detection systems, particularly those utilising machine learning and natural language processing (NLP), are essential for identifying and mitigating the impact of misinformation in real time [6, 7].

The support vector machine (SVM) and Multinomial Naïve Bayes are among the traditional machine learning methods. The machine

learning, like Multinomial Naïve Bayes, Logistic Regression, and Random Forest, provides interpretable models that present considerable benefits in real-time systems [8, 9]. Conversely, deep learning methods, particularly transformer-based models such as BERT, achieve superior accuracy in natural language processing (NLP). However, this necessitates substantial computational resources and extended training durations. As a result, this may impact real-time detection, which is essential in these types of systems [10].

Despite the growing body of work in fake news detection, gaps remain. Many existing models are either domain-specific, tightly coupled to social media features, or overly dependent on resource-heavy architectures. This study addresses these limitations by proposing an efficient, interpretable, and domain-agnostic detection framework grounded in classical ML techniques.

Although the study is motivated by misinformation as a soft-power instrument, the WELFake dataset provides only binary credibility labels (Real vs. Fake) and does not annotate soft-power intent. Accordingly, the supervised objective and reported results are strictly binary fake-news detection. References to "soft power" are therefore

motivational and interpretive, not a claim of supervised soft-power intent classification [11, 12].

The urgency of this research stems from the escalating use of misinformation as a tool for psychological and sociopolitical disruption. As false narratives are increasingly weaponised to manipulate elections, weaken public health compliance, and destabilise institutional trust, scalable and automated detection systems become a societal necessity. Manual content moderation and traditional fact-checking are insufficient to respond to the speed and scale of these campaigns. Thus, this study seeks to enhance the reliability and accessibility of ML-based detection tools—especially those suitable for deployment in real-time or constrained environments.

Problem statement and scope. Despite the rapid progress in automated fake-news detection, the core problem remains difficult in practice: deceptive articles often mimic credible journalistic language and differ from real reporting through subtle framing, selective emphasis, or persuasive tone rather than obvious lexical errors. Therefore, the key challenge is to build a detection approach that is both accurate and reliable while remaining lightweight and interpretable enough for real-time or resource-constrained deployment. In this work, the task is defined strictly as binary credibility classification (Real vs. Fake) because WELFake provides only these labels; “soft power” is used as a socio-security motivation for why fake news matters, not as an additional supervised label or claim of intent prediction [11, 12].

This paper’s main contributions are as follows:

- It motivates fake-news detection within a soft-power influence context and clarifies that, in this study, the supervised task is restricted to binary Real/Fake credibility classification because WELFake does not provide influence-intent labels.
- It presents a reproducible and leakage-safe TF-IDF-based pipeline for fake-news classification, including explicit preprocessing, stratified hold-out evaluation, and transparent reporting of model settings.
- It empirically demonstrates that LinearSVC achieves 95.65% accuracy on the WELFake dataset under a leakage-safe protocol, while providing strong precision, recall, and F1-score balance across both classes.
- It provides a practical and interpretable benchmark for fake-news detection in resource-constrained or real-time settings by comparing classical machine-learning models under a unified evaluation framework.

2.Related Work

In 2023, Dementieva et al. proposed Multiverse: a cross-lingual evidence aggregation method to improve fake news detection across languages by combining signals from multiple linguistic views [13]. While effective at boosting robustness in multilingual settings, Multiverse adds complexity by requiring external cross-language evidence and alignment. In contrast, in this work, the proposed method is single-language, lightweight, and domain-agnostic, using a standard, reproducible TF-IDF representation with classical classifiers (e.g., LinearSVC) to achieve high accuracy without multilingual evidence fusion.

In 2023, Mohawesh et al. developed a multilingual deep learning framework for fake news detection that leverages transfer learning to adapt across languages and domains, particularly in the COVID-19 misinformation context [14]. Their approach depends on deep architectures and substantial pretraining to handle language variation, which increases computational cost and reduces interpretability. On the other hand, our framework avoids heavy deep models, delivering comparable practical utility through an efficient preprocessing pipeline and interpretable classical classifiers, making it better suited for real-time deployment and environments lacking extensive multilingual training data.

Also in 2023, the FakeWatch ElectionShield project introduced a benchmarking platform and dataset tailored to election-related fake news, combining classical and modern models to evaluate effectiveness in a high-stakes topical domain [15]. Although it demonstrates that structured evaluation and hybrid modelling help in narrowly defined contexts, its focus is limited to election content and often relies on curated, event-specific ground truth. Our work, otherwise, generalises beyond a single topical domain, using a unified, scalable pipeline that does not require specialised datasets per event while maintaining balanced detection performance.

In 2023, Kumarage et al. presented J-Guard, which uses journalistic stylistic cues to guide adversarial robust detection of AI-generated or manipulated news [16]. Their method improves robustness by incorporating high-level style constraints, but it increases model complexity and assumes access to reliable stylistic priors. Our proposed approach emphasises simplicity and interpretability. Furthermore, it is capable of detecting subtle misinformation directly from textual patterns via TF-IDF and margin-based classification without needing external stylistic guidance, while remaining extensible for future robustness enhancements.

In 2024, Muñoz, Pérez-Rosas, and Mihalcea investigated explainable fake news detection by exploiting content characteristics—linguistic, affective, and cognitive features—to differentiate real from fake news [17]. Their feature-rich explainability adds transparency but involves complex hand-crafted or derived feature sets that can be brittle across domains. Compared with the works in this paper, the proposed frameworks achieve explainability indirectly through inherently interpretable classifiers (like LinearSVC) on top of a streamlined, automated TF-IDF pipeline. This led to avoiding brittle feature engineering while still enabling insight into decision rationale. Alghamdi et al. presented a hybrid summarisation and transformer solution to noisy multilingual input for low-resource language false news detection in 2024 [18]. Transformer backbones and language-specific adaptation techniques enable detection in scarce-resource languages, but they add computing overhead and deployment complexity. Transformer-free, model-light, and homogeneous across domains, our solution is more practical for limited situations without compromising monolingual data detection quality.

In 2024, Almandouh et al. explored ensemble strategies that fuse transformer embeddings with classical models for Arabic fake news detection, achieving high accuracy through complex model stacking [19]. Their ensemble design, while powerful, reduces transparency and inflates inference cost. In contrast, our solution delivers strong performance with a single classical classifier, preserving

interpretability and minimising runtime overhead, which simplifies maintenance and scaling.

Also in 2024, Maham et al. introduced ANN: Adversarial News Net, an adversarial trained framework to harden fake news classifiers against perturbations and deceptive manipulations [20]. Their adversarial training regime improves robustness but demands careful attack modelling and increases training complexity. Our approach, on the other hand, prioritises immediate applicability: using a fixed, explicitly specified TF-IDF representation with LinearSVC provides strong baseline detection performance without requiring an adversarial setup, while remaining compatible with future robustness extensions.

Finally, Panchendrarajan et al. (2024) surveyed the space of claim detection for automated fact-checking, emphasising multilingual, verifiability-aware systems as precursors to misinformation identification [21]. While comprehensive, claim detection is a complementary upstream task, rather than a direct classifier of fake news, and many surveyed pipelines still depend on multi-stage complexity. On the other hand, when compared with this paper's work, the proposed framework operates as a simpler, faster, and more direct end-to-end detection mechanism. Furthermore, it focused on classification, while still aligning with the broader goal of language-agnostic integrity verification.

The proposed framework advances beyond these prior studies through its combination of explicitly specified feature extraction, class-wise evaluation, and interpretability, delivering an efficient and scalable fake-news detection system suitable for real-world, real-time deployment.

3. Methodology

3.1. Overview and Task Definition

This study addresses fake-news detection as a binary supervised text classification problem, where each news article is labelled as Real (0) or Fake (1). The objective is a controlled comparison of classical machine-learning baselines under a single, leakage-safe preprocessing and evaluation protocol.

3.2. Dataset, Cleaning, and Duplicate Control

The Experiments in this work have used the WELFake dataset, which provides binary fake/real labels for textual news content [11, 12].

First, before modelling, the dataset is cleaned to prevent noisy inputs and leakage: rows with missing text/labels are removed; texts are cast to strings; empty/whitespace-only texts are discarded; and duplicate texts are removed. All cleaning and deduplication are performed before the train-test split to prevent duplicate leakage across partitions. The held-out test split label distribution is reported in Table 1.

Table 1. Testing Dataset

Label	Count	Percentage
Real	5,619	44.8%
Fake	6,924	55.2%
Total	12,543	100%

3.3. Text Representation with TF-IDF

Each article was represented using term frequency-inverse document frequency (TF-IDF), a standard weighting scheme that emphasizes terms that are frequent within a document but less

frequent across the corpus. TF-IDF features were computed using a standard vectorizer implementation [22].

The main representation used word unigrams, with text lowercasing enabled. English stop-words were removed, and extremely common terms were filtered by discarding tokens that appeared in more than 70% of documents. No stemming or lemmatization was applied.

Tokenization and normalization behaviour: Tokens were defined as sequences of at least two-word characters (letters, digits, or underscores). Under this rule, most standalone punctuation marks and emojis are excluded during tokenization, while numerals can remain when they appear as part of valid tokens (for example, "2024").

The full TF-IDF configuration used in the experiments is reported in Table 2. The overall pipeline (cleaning → TF-IDF → classification → evaluation) is summarized in Figure 1.

TF-IDF definition (Word-friendly):

Let $tf(t, d)$ be the frequency of the term t in document d , while $df(t)$ the number of documents containing t , and N the total number of documents. Then:

1. $idf(t) = \ln((1 + N) / (1 + df(t))) + 1$
2. $tfidf(t, d) = tf(t, d) \times idf(t)$

Table 2. Word-level TF-IDF configuration (scikit-learn)

Parameter	Value used
Unit of analysis	Word
N-gram range	Unigrams only
Lowercasing	Enabled
Stop-word removal	English stop-word list
Maximum document frequency	0.70
Minimum document frequency	1
Maximum number of features	No explicit limit
Token definition	Tokens must contain at least two word characters.
Vector normalization	L2 normalization
Inverse document frequency	Enabled
IDF smoothing	Enabled
Sublinear term-frequency scaling	Not used
Accent stripping	Not used
Unit of analysis	Word

3.4. Train-Test Split, Leakage Control, and Reproducibility

A stratified hold-out split was used with 80% of the data for training and 20% for testing, preserving the class proportions in both partitions. A fixed seed of 42 was used to ensure reproducibility of the split.

To prevent feature leakage, the TF-IDF representation (including the vocabulary and inverse-document-frequency statistics) was learned using only the training texts. The held-out test texts were then represented using the TF-IDF mapping learned from the training partition only. Each model was trained once on the training split and evaluated once on the held-out test split[23].

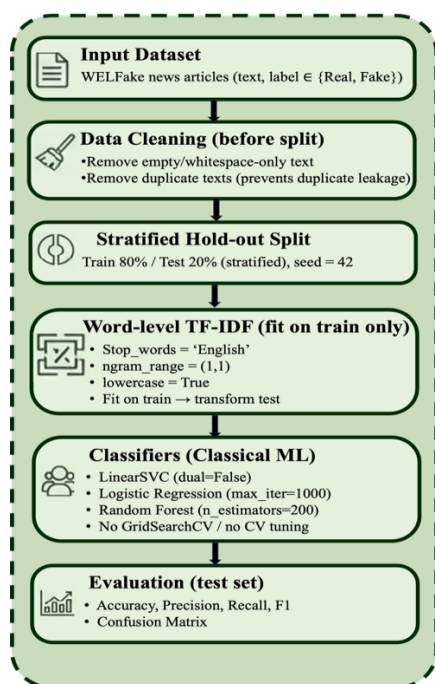


Figure 1. The proposed Model Architecture

3.4.1 Class Imbalance Handling

The WELFake corpus exhibits moderate class imbalance in the held-out test split (Fake: 55.2%, Real: 44.8%; Table 1). No over-sampling or under-sampling was applied at any stage, and no class reweighting was used during training. Instead, class proportions were preserved through stratified partitioning, and performance was assessed using class-wise precision, recall, and F1-score, in addition to confusion-matrix analysis. Under this setting, any potential effect of the imbalance on the LinearSVC separating margin is evaluated indirectly by inspecting whether errors concentrate disproportionately in one class (false positives vs. false negatives) and by verifying balanced per-class performance rather than relying on accuracy alone [24].

3.5. Classification Algorithms and Justifications

Four classical machine-learning classifiers were evaluated in this work. These machine learning techniques are widely used baselines for text classification and remain attractive in practical fake-news detection settings due to their efficiency, transparency, and well-understood behaviour on sparse high-dimensional representations such as TF-IDF [25]. In this work, the classifiers were selected to represent complementary modelling assumptions: (i) maximum-margin linear separation (LinearSVC), (ii) probabilistic linear modelling (Logistic Regression), (iii) generative probabilistic modelling under conditional independence (Multinomial Naïve Bayes), and (iv) non-linear ensemble learning through decision-tree aggregation (Random Forest). This set provides a balanced and interpretable comparison under a unified feature space and evaluation protocol. Enabling conclusions about which modelling family aligns best with TF-IDF features for the WEL Fake task.

3.5.1. Linear Support Vector Classification (Linear SVC)

Linear support vector classification is a discriminative learning approach that seeks a linear separating boundary between classes while maximizing the margin, i.e., the geometric distance between the boundary and the closest training samples of each class [26]. In text classification, TF-IDF produces extremely high-dimensional

sparse vectors in which many terms occur rarely, and only a small subset of features is active per document. Maximum-margin learning is well suited to this regime because it focuses on the most informative training instances near the decision boundary and tends to generalize well even when the number of features is very large. The linear decision rule also aligns naturally with TF-IDF: when data are embedded in a high-dimensional space, documents often become more linearly separable than they would be in low-dimensional representations. This makes a linear margin-based separator highly competitive.

From a practical standpoint, LinearSVC is attractive for fake-news detection because it achieves high accuracy with low inference cost. Once trained, prediction involves a sparse dot product between a document vector and a learned weight vector, which is computationally efficient. This property makes it appropriate for real-time or large-scale screening applications where many documents must be classified quickly. Moreover, even though the classifier is not probabilistic by default, its learned weights still provide interpretability: high-magnitude weights indicate terms that strongly influence the decision boundary, which can be inspected to understand the lexical tendencies associated with each class under TF-IDF.

3.5.2. Logistic Regression

Logistic Regression is a discriminative probabilistic classifier that models the conditional probability of class membership given the input features [27]. It assumes that the log-odds of the class can be represented as a linear function of the feature vector. In the context of TF-IDF features, this results in a linear decision boundary similar in form to linear SVMs, but with a probability-based interpretation. The probabilistic perspective is useful in many credibility-detection settings because it enables threshold adjustment: an application can choose a conservative threshold to minimize false positives, or a more aggressive threshold to maximize recall for fake-news detection, depending on policy priorities.

The model is typically trained by maximizing a regularized likelihood objective. Regularization is critical when handling TF-IDF because the feature space contains many dimensions, and without regularization, the model could overfit to spurious term correlations. L2 regularization (ridge penalty) is a standard choice because it discourages overly large weights and promotes stability. In the proposed framework, training was allowed up to 1000 iterations to ensure convergence, and a fixed seed was retained for configuration consistency across experiments. The optimization used a deterministic quasi-Newton procedure, which typically yields stable results and avoids dependence on stochastic randomness.

Compared with LinearSVC, Logistic Regression often performs similarly on sparse text features because both are linear separators [27]. Differences typically arise from their objective functions: margin-based objectives emphasize maximizing separation near the boundary, whereas logistic objectives emphasize probability fit. In practice, Logistic Regression can be slightly more sensitive to overlapping class distributions when the separation is subtle, but its probability outputs remain valuable for interpretability and operational decision design.

3.5.3. Multinomial Naïve Bayes

Multinomial Naïve Bayes is a generative probabilistic classifier derived from Bayes' theorem. It models each class as a distribution over features and assumes conditional independence of features given the class label. For text classification, the multinomial variant is well known because it fits naturally with token-count representations and can also be applied to TF-IDF style nonnegative feature vectors as a baseline [28]. The strength of Naïve Bayes lies in its simplicity and speed: training involves estimating class-conditional feature statistics and class priors, and prediction involves combining these quantities to compute posterior scores.

A key element in Naïve Bayes for text is smoothing, which prevents zero-probability issues when a term does not appear in the training data for a particular class [28]. During evaluation, additive smoothing with alpha equal to 1.0 was used. Also, the class priors were estimated from the training distribution. These default settings are widely used as a baseline configuration and are sufficient to evaluate the general suitability of the model family under TF-IDF.

Despite its efficiency, Naïve Bayes can underperform on fake-news detection when deception cues depend on combinations of terms, contextual phrasing, or stylistic patterns that violate the independence assumption. Fake news narratives often reuse credible vocabulary, and the difference may be subtle and dependent on multi-term framing rather than independent word evidence. As a result, Naïve Bayes may struggle to represent the interaction effects needed to separate classes in ambiguous cases.

3.5.4. Random Forest

Random Forest is an ensemble method that aggregates the predictions of many decision trees to improve robustness and reduce variance. Each tree is trained on a bootstrap sample of the training data, and at each split only a random subset of features is considered [29]. This randomness decorrelates trees and helps prevent overfitting. Random Forest can represent non-linear decision boundaries and capture feature interactions that linear models cannot explicitly model [29].

In text classification with TF-IDF, Random Forest has both potential strengths and limitations. Its strength is modelling non-linear interactions: certain combinations of terms may be predictive in ways that are not well expressed by linear weights. However, TF-IDF yields extremely high-dimensional sparse inputs, which can be challenging for tree-based methods. Trees must search for informative splits among a very large number of features, many of which are rarely active. This can make training more expensive and may reduce the advantage of non-linearity compared to tasks where features are compact and dense. Additionally, interpretability is less direct than in linear models because the decision logic is distributed across many trees, although feature-importance measures can still provide a coarse indication of influential terms.

In our experiments, the Random Forest baseline used 200 trees, was trained with parallel computation, and used a fixed seed to support reproducibility. Trees were grown without an explicit depth limit under standard split and leaf constraints, with bootstrap sampling enabled.

3.5.5. Model Selection Criterion

Model selection was based on held-out test performance reported in Section 4 (Table 2 and Figures 2–3), using accuracy and class-wise

precision/recall/F1 as the primary criteria. The selected classical baseline is the model with the best overall test performance under the leakage-safe protocol described in Section 3.4. Table 3 summarizes the key hyperparameters used for each classifier

Table 3. classifier hyperparameters summary

Classifier	Reported configuration (explicitly set)	Defaults reported (library defaults used)
LinearSVC	Dual formulation disabled (primal optimization); fixed random seed (42).	Regularization strength C = 1.0; loss = squared hinge; penalty = L2; maximum iterations = 1000.
Logistic Regression	Maximum iterations = 1000; fixed random seed (42).	Solver = L-BFGS; penalty = L2; regularization strength C = 1.0.
Multinomial Naïve Bayes	No overrides (default configuration).	Additive smoothing alpha = 1.0; class priors estimated from the training data.
Random Forest	Number of trees = 200; parallel training across CPU cores; fixed random seed (42).	Maximum depth = unlimited; split criterion = Gini; minimum samples to split = 2; minimum samples per leaf = 1; bootstrap sampling enabled; feature subset per split = library default for classification.

3.5.6. Shallow Neural Baseline: LSTM

To complement the classical baselines, we also evaluated a shallow neural network baseline based on the Long Short-Term Memory (LSTM) architecture. LSTM networks are designed to model sequential dependencies by using gated memory cells that mitigate the vanishing-gradient limitation of standard recurrent neural networks, making them suitable for natural language sequences where contextual information may span multiple tokens [30, 31].

For this baseline, each news article was converted into a sequence of integer token indices using a fixed vocabulary derived from the training set. To enable batch training, sequences were standardized to a fixed length by truncating longer texts and padding shorter ones. The model consists of an embedding layer that maps tokens into dense vectors, followed by a single LSTM layer that produces a contextual document representation, and a final sigmoid output unit for binary classification (Fake vs. Real). Training was performed using binary cross-entropy loss and an adaptive gradient optimizer. All LSTM preprocessing choices and hyperparameter settings used in the experiments are reported in Table 4 to ensure full reproducibility. The evaluation followed the same stratified train/test split used throughout the study (Table 1), and the final metrics were computed only on the held-out test set.

The LSTM baseline was trained for three epochs as a lightweight comparative reference rather than a fully optimized deep-learning model; preliminary validation behaviour stabilized early, and extensive deep-model tuning was intentionally kept outside the scope of this classical-ML-focused benchmark.

Table 4. LSTM baseline configuration used

Component / Setting	Value
Tokenization	word-level tokenizer; vocabulary built from training set
Vocabulary size	50,000 most frequent tokens
Sequence length	250 tokens (padding/truncation)

Component / Setting	Value
Embedding dimension	100
Recurrent layer	single-layer LSTM
Hidden units	100
Dropout	0.2
Output layer	1 neuron + sigmoid
Loss function	binary cross-entropy
Optimizer	Adam
Learning rate	0.001
Batch size	64
Epochs	3

4. Experimental Results and Discussion

This section presents a comparative evaluation of the selected classifiers for binary fake-news detection on the WELFake dataset under a stratified hold-out testing protocol. After cleaning the data and removing duplicates (both performed prior to splitting to prevent content repetition across partitions), the dataset was divided into 80% training and 20% testing, while preserving the class proportions through stratification. A fixed seed was used to ensure that the split is reproducible and that subsequent experimental runs are directly comparable. The held-out test split used for reporting in this section contains 12,543 samples, and its class distribution is summarized in Table 1.

The results are reported using standard classification metrics alongside confusion-matrix analysis. This combination is important because it provides both (i) a compact numerical summary of performance, and (ii) an interpretable view of the specific types of errors each model makes. In fake-news detection, this distinction matters: a model can achieve strong overall accuracy while still producing problematic errors in one class (for example, over-flagging real news as fake, or missing fake items and labelling them real). Therefore, the discussion below emphasizes not only headline accuracy, but also class-wise precision and recall, and how they relate to practical reliability.

4.1. Evaluation Metrics

Performance was measured on the test set using accuracy and class-wise precision, recall, and F1-score for both classes (Fake and Real). The definitions are [32]:

$$P = \frac{TP}{TP + FP} \quad \dots \dots (1)$$

$$R = \frac{TP}{TP + FN} \quad \dots \dots (2)$$

$$F1 = \frac{2 \times P \times R}{(P + R)} \quad \dots \dots (3)$$

$$Accuracy = \frac{(TP + TN)}{(TP + TN + FP + FN)} \quad \dots \dots (4)$$

Where TP denotes true positives, FP false positives, FN false negatives, and TN true negatives. In this study, the “positive” class is Fake, so TP corresponds to Fake news predicted as Fake correctly, FP corresponds to Real news incorrectly predicted as Fake, FN corresponds to Fake news incorrectly predicted as Real, and TN corresponds to Real news correctly predicted as Real.

In addition to class-wise scores, macro averages (unweighted mean across classes) and weighted averages (mean weighted by class

support) were computed to summarize overall behaviour under the observed class distribution (Table 1).

4.2. Quantitative Comparison on the Test Set

Table 5 reports the full test-set results for all evaluated models. Among the classical baselines, LinearSVC achieved the best overall performance, followed by Logistic Regression and Random Forest. The LSTM baseline provides a shallow neural reference; it slightly exceeds Multinomial Naïve Bayes in overall accuracy but remains substantially below the stronger linear and ensemble baselines on this dataset.

Table.5. Test-set performance on WELFake

Model	Accuracy	Precision (Fake)	Recall (Fake)	F1-Score (Fake)	Precision (Real)	Recall (Real)	F1-Score (Real)
Multinomial Naïve Bayes	0.8687	0.86	0.87	0.87	0.87	0.86	0.87
LSTM	0.8744	0.9231	0.8427	0.8811	0.8250	0.9135	0.8670
Logistic Regression	0.9409	0.95	0.93	0.94	0.93	0.96	0.94
Random Forest	0.9324	0.94	0.92	0.93	0.92	0.94	0.93
LinearSVC	0.9565	0.97	0.94	0.96	0.95	0.97	0.96

As visualized in Figure 2, LinearSVC provides the highest accuracy. More importantly, its class-wise results are well balanced: the model maintains high precision for Fake while sustaining strong recall for Fake, and similarly strong performance for Real. This balance indicates that the model is not simply exploiting moderate label imbalance; it separates the classes effectively in the TF-IDF space. Logistic Regression performs closely behind LinearSVC, reflecting the suitability of linear decision functions for TF-IDF representations. Random Forest performs competitively but slightly lower, which is consistent with tree ensembles being less naturally aligned with extremely high-dimensional sparse feature spaces. Multinomial Naïve Bayes is efficient and stable but shows reduced performance, which is expected when conditional-independence assumptions cannot fully capture context-dependent cues in deceptive text.

Transformer-based baselines (e.g., DistilBERT) were not included in this experimental comparison due to their higher computational cost and reduced transparency at inference time; instead, we report a shallow neural baseline (LSTM) and focus on lightweight, interpretable models suitable for resource-constrained or real-time deployment, leaving transformer comparisons for future work. For quantitative context, the DistilBERT baseline in our implementation uses approximately 66 million parameters, whereas the LSTM baselines used here contain approximately 5.08 million parameters (PyTorch LSTM) and 0.82 million parameters (Keras LSTM), respectively. This substantial difference in model scale supports the decision to prioritize interpretable and computationally efficient baselines in the present study.

4.3. Confusion Matrix Interpretation (Error Analysis)

Figure 3 reports the confusion matrix for the best-performing classical model (LinearSVC) and provides insight beyond overall accuracy. In fake-news detection, the type of error is important: false positives (Real $\rightarrow$ Fake) may reduce trust by incorrectly flagging legitimate content, whereas false negatives (Fake $\rightarrow$ Real) allow misinformation to pass as credible. Therefore, the confusion matrix is interpreted together with the class-wise metrics in Table 5.

The raw confusion-matrix counts are as follows: 5392 Real articles were correctly classified as Real, 229 Real articles were misclassified as Fake, 249 Fake articles were misclassified as Real, and 6673 Fake articles were correctly classified as Fake. These values are consistent with the class-wise results in Table 5 and show that the model maintains a balanced error profile across both classes.

For the Fake class, LinearSVC achieves high precision (0.97) and high recall (0.94), indicating that most items predicted as fake are indeed fake, while the majority of fake articles are successfully detected. For the Real class, the model also performs strongly (precision = 0.95, recall = 0.97), suggesting that the classifier is not merely benefiting from the moderate class imbalance. Instead, it separates the two classes effectively in the TF-IDF feature space.

The remaining errors likely correspond to borderline cases in which linguistic cues overlap. Some fake articles imitate credible reporting style through formal tone, plausible structure, and named entities, leading to false negatives. Conversely, some real articles use urgent or persuasive framing, increasing the likelihood of being flagged as fake. Overall, Figure 3 indicates that LinearSVC errors arise mainly from textual ambiguity rather than systematic bias toward either class.

4.4. Decision Rule for LinearSVC

For LinearSVC, labels are assigned using the signed distance from the separating hyperplane. A zero threshold separates the two classes, and no probability calibration was applied:

Overall accuracy is computed as the sum of all true positives across classes divided by the total number of samples. Macro averages are computed as the unweighted mean across classes, while weighted averages take into account the support (sample count) per class.

4.5. Reporting Scope and Statistical Testing

The analysis in this study reports point-estimate performance on a stratified held-out test set using accuracy, class-wise precision, recall, F1-score, and confusion-matrix inspection. This reporting scope is appropriate for the present objective, which is comparative benchmarking under a reproducible and leakage-safe protocol rather than formal inferential hypothesis testing. In addition, the held-out test set is relatively large (12,543 samples), which provides a stable basis for descriptive comparison and reduces concerns about extreme variance in single-split evaluation. Accordingly, the revised manuscript focuses on transparent performance reporting across models using the same preprocessing, split, and evaluation conditions. Statistical significance procedures such as confidence intervals, McNemar's test, or bootstrap resampling are not reported here because they were not implemented as part of the current experimental pipeline. These analyses may be incorporated in future work once repeated runs or resampling-based evaluation is added in a standardized manner.

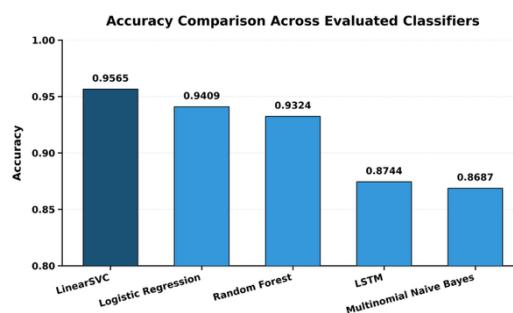


Figure 2. Model Accuracy

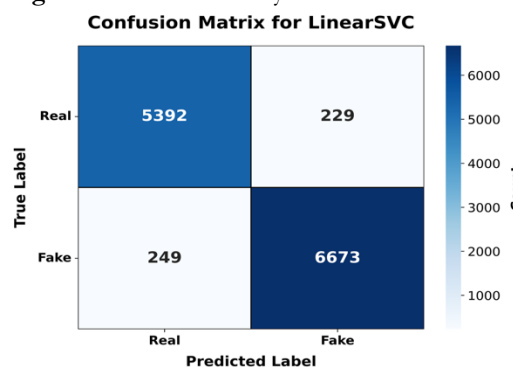


Figure 3. Confusion Matrix

4.6. Soft-Power Interpretation

Although WELFake does not annotate soft-power intent, the soft-power framing remains useful for interpreting why certain borderline errors occur in purely textual credibility classification. Soft-power narratives typically prioritize persuasion over verifiable detail, often using emotionally charged language, appeals to identity, urgency cues, selective emphasis, and confident tone to steer beliefs and trust. In practice, such rhetorical features can appear in both deceptive and credible reporting, which explains why some Real articles may be flagged as Fake when they adopt highly urgent or persuasive framing (false positives). Conversely, some Fake articles mimic legitimate news conventions by using formal structure, neutral vocabulary, named entities, and plausible factual scaffolding, which can cause them to be predicted as Real (false negatives). The near-symmetry of the two error types in the confusion matrix suggests that the classifier does not simply exploit label prevalence but instead confronts a genuine overlap zone where “influence-style” writing and standard reporting partially intersect. This overlap is consistent with real-world disinformation strategies, where credibility mimicry is deliberately used to reduce detectability. Therefore, the role of the soft-power perspective in this paper is interpretive: it highlights the kinds of linguistic framing that make fake-news detection challenging even when accuracy is high. Future work can strengthen this connection by introducing a reproducible annotation protocol for influence intent or by evaluating the pipeline on datasets that explicitly label propaganda or persuasive intent.

5. Conclusion

In this study, a comparative evaluation of classical machine-learning algorithms for fake-news detection under a unified and leakage-safe experimental protocol was conducted. Four widely used classifiers—LinearSVC, Logistic Regression, Multinomial Naïve Bayes, and Random Forest—were assessed on the WELFake dataset using

consistent TF-IDF-based text representation and the same stratified hold-out setting.

Among the tested models, LinearSVC demonstrated superior performance, achieving the highest accuracy (95.65%) and outperforming the other algorithms in terms of precision, recall, and F1-score. These results indicate that linear margin-based classification is highly effective for high-dimensional sparse textual representations, particularly when combined with TF-IDF features. While Logistic Regression and Random Forest also showed strong performance, they remained slightly below LinearSVC in the overall balance between predictive accuracy, interpretability, and computational efficiency. Naïve Bayes, although fast and lightweight, was less effective in handling semantically nuanced misinformation. Beyond classification accuracy, the findings also highlight the broader sociopolitical relevance of fake-news detection, where misinformation may function as an influence instrument capable of shaping public perception and weakening trust in institutions. However, the present experiments address binary credibility classification only, since WELFake does not provide soft-power or influence-intent labels. Therefore, while the study is motivated by the role of misinformation in soft-power contexts, it does not claim supervised detection of influence intent. The ability to automatically detect and counter such content is crucial for maintaining public trust and democratic resilience. Future work may extend this framework toward intent-aware detection through dedicated annotation, richer contextual features, and modern transformer-based or multimodal baselines.

6. References

- [1] B. Rød, C. Pursiainen, and N. Eklund, "Combatting Disinformation – How Do We Create Resilient Societies? Literature Review and Analytical Framework," *European Journal for Security Research*, 2025, <https://doi.org/10.1007/s41125-025-00105-4>.
- [2] P. N. Vasist and S. Krishnan, "Country branding in post-truth Era: A configural narrative," *Journal of Destination Marketing & Management*, vol. 32, p. 100854, 2024, <https://doi.org/10.1016/j.jdmm.2024.100854>.
- [3] J. Arayankalam and S. Krishnan, "Relating foreign disinformation through social media, domestic online media fractionalization, government's control over cyberspace, and social media-induced offline violence: Insights from the agenda-building theoretical perspective," *Technological Forecasting and Social Change*, vol. 166, p. 120661, 2021, <https://doi.org/10.1016/j.techfore.2021.120661>.
- [4] R. Marigliano, L. H. X. Ng, and K. M. Carley, "Analyzing digital propaganda and conflict rhetoric: a study on Russia's bot-driven campaigns and counter-narratives during the Ukraine crisis," *Social Network Analysis and Mining*, vol. 14, no. 1, 2024, <https://doi.org/10.1007/s13278-024-01322-w>.
- [5] M. Soprano et al., "Cognitive Biases in Fact-Checking and Their Countermeasures: A Review," *Information Processing & Management*, vol. 61, no. 3, p. 103672, 2024, <https://doi.org/10.1016/j.ipm.2024.103672>.
- [6] R. Ş. Emil and B. Remus, "A Review of Automatic Fake News Detection: From Traditional Methods to Large Language Models," *Future Internet*, vol. 17, no. 10, p. 435, 2025, <https://doi.org/10.3390/fi17100435>.
- [7] E. S. Albtoush, K. H. Gan, and S. A. A. Alrababa, "Fake news detection: state-of-the-art review and advances with attention to Arabic language aspects," *PeerJ Computer Science*, vol. 11, p. e2693, 2025, <https://doi.org/10.7717/peerj-cs.2693>.
- [8] J. Alghamdi, Y. Lin, and S. Luo, "A Comparative Study of Machine Learning and Deep Learning Techniques for Fake News Detection," *Information*, vol. 13, no. 12, p. 576, 2022. [Online]. Available: <https://www.mdpi.com/2078-2489/13/12/576>.
- [9] P. Linardatos, V. Papastefanopoulos, and S. Kotsiantis, "Explainable AI: A Review of Machine Learning Interpretability Methods," *Entropy*, vol. 23, no. 1, p. 18, 2021. [Online]. Available: <https://www.mdpi.com/1099-4300/23/1/18>.
- [10] E. Strubell, A. Ganesh, and A. McCallum, "Energy and Policy Considerations for Deep Learning in NLP," Florence, Italy, July 2019: Association for Computational Linguistics, in *Proceedings of the 57th Annual Meeting of the Association for Computational Linguistics*, pp. 3645-3650, <https://doi.org/10.18653/v1/P19-1355>.
- [11] P. K. Verma, P. Agrawal, and R. Prodan, "WELFake dataset for fake news detection in text data", Zenodo, 2021, <https://doi.org/10.5281/zenodo.4561253>.
- [12] P. K. Verma, P. Agrawal, I. Amorim, and R. Prodan, "WELFake: Word Embedding Over Linguistic Features for Fake News Detection," *IEEE Transactions on Computational Social Systems*, vol. 8, no. 4, pp. 881-893, 2021, <https://doi.org/10.1109/TCSS.2021.3068519>.
- [13] D. Dementieva, M. Kuimov, and A. Panchenko, "Multiverse: Multilingual Evidence for Fake News Detection," *Journal of Imaging*, vol. 9, no. 4, p. 77, 2023. <https://www.mdpi.com/2313-433X/9/4/77>.
- [14] R. Mohawesh, S. Maqsood, and Q. Althebyan, "Multilingual deep learning framework for fake news detection using capsule neural network," *Journal of Intelligent Information Systems*, vol. 60, no. 3, pp. 655-671, 2023, <https://doi.org/10.1007/s10844-023-00788-y>.
- [15] D. K. Sharma and S. Garg, "IFND: a benchmark dataset for fake news detection," *Complex & Intelligent Systems*, vol. 9, no. 3, pp. 2843-2863, 2023, <https://doi.org/10.1007/s40747-021-00552-1>.
- [16] T. Kumarage et al., "J-Guard: Journalism Guided Adversarially Robust Detection of AI-generated News," Nusa Dua, Bali, November 2023: Association for Computational Linguistics, in *Proceedings of the 13th International Joint Conference on Natural Language Processing and the 3rd Conference of the Asia-Pacific Chapter of the Association for Computational Linguistics (Volume 1: Long Papers)*, pp. 484-497, <https://doi.org/10.18653/v1/2023.ijcnlp-main.32>.
- [17] S. Muñoz and C. Á. Iglesias, "Exploiting Content Characteristics for Explainable Detection of Fake News," *Big Data and Cognitive Computing*, vol. 8, no. 10, p. 129, 2024. [Online]. Available: <https://www.mdpi.com/2504-2289/8/10/129>.

- [18] J. Alghamdi, Y. Lin, and S. Luo, "Enhancing hierarchical attention networks with CNN and stylistic features for fake news detection," *Expert Systems with Applications*, vol. 257, p. 125024, 2024, <https://doi.org/10.1016/j.eswa.2024.125024>
- [19] M. E. Almandouh, M. F. Alrahmawy, M. Eisa, M. Elhoseny, and A. S. Tolba, "Ensemble based high performance deep learning models for fake news detection," *Scientific Reports*, vol. 14, no. 1, 2024, <https://doi.org/10.1038/s41598-024-76286-0>
- [20] S. Maham, A. Tariq, M. U. G. Khan, F. S. Alamri, A. Rehman, and T. Saba, "ANN: adversarial news net for robust fake news classification," *Scientific Reports*, vol. 14, no. 1, 2024, <https://doi.org/10.1038/s41598-024-56567-4>
- [21] R. Panchendrarajan and A. Zubiaga, "Claim detection for automated fact-checking: A survey on monolingual, multilingual and cross-lingual research," *Natural Language Processing Journal*, vol. 7, p. 100066, 2024, <https://doi.org/10.1016/j.nlp.2024.100066>
- [22] G. Salton and C. Buckley, "Term-weighting approaches in automatic text retrieval," *Information Processing & Management*, vol. 24, no. 5, pp. 513-523, 1988/01/01/ 1988, [https://doi.org/10.1016/0306-4573\(88\)90021-0](https://doi.org/10.1016/0306-4573(88)90021-0)
- [23] S. Kapoor and A. Narayanan, "Leakage and the reproducibility crisis in machine-learning-based science," *Patterns*, vol. 4, no. 9, 2023, <https://doi.org/10.1016/j.patter.2023.100804>
- [24] P. Billion Polak, J. D. Prusa, and T. M. Khoshgoftaar, "Low-shot learning and class imbalance: a survey," *Journal of Big Data*, vol. 11, no. 1, p. 1, 2024, <https://doi.org/10.1186/s40537-023-00851-z>
- [25] K. Taha, P. D. Yoo, C. Yeun, D. Homouz, and A. Taha, "A comprehensive survey of text classification techniques and their research applications: Observational and experimental insights," *Computer Science Review*, vol. 54, p. 100664, 2024, <https://doi.org/10.1016/j.cosrev.2024.100664>
- [26] C. Cortes and V. Vapnik, "Support-vector networks," *Machine Learning*, vol. 20, no. 3, pp. 273-297, 1995, <https://doi.org/10.1007/bf00994018>
- [27] G. James, D. Witten, T. Hastie, and R. Tibshirani, "An introduction to statistical learning: with applications in R", Springer, 2013.
- [28] C. D. Manning, P. Raghavan, and H. Schütze, *Introduction to Information Retrieval*. Cambridge: Cambridge University Press, 2008.
- [29] L. Breiman, "Random Forests," *Machine Learning*, vol. 45, no. 1, pp. 5-32, 2001, <https://doi.org/10.1023/A:1010933404324>
- [30] G. Van Houdt, C. Mosquera, and G. Nápoles, "A review on the long short-term memory model," *Artificial Intelligence Review*, vol. 53, no. 8, pp. 5929-5955, 2020, <https://doi.org/10.1007/s10462-020-09838-1>
- [31] S. Hochreiter and J. Schmidhuber, "Long Short-Term Memory," *Neural Computation*, vol. 9, no. 8, pp. 1735-1780, 1997, <https://doi.org/10.1162/neco.1997.9.8.1735>
- [32] M. Sokolova and G. Lapalme, "A systematic analysis of performance measures for classification tasks," *Information Processing & Management*, vol. 45, no. 4, pp. 427-437, 2009, <https://doi.org/10.1016/j.ipm.2009.03.002>