electronics

Article

A Weighted-Distance Ensemble Learning Method for Arabic Fake News Detection

Dhafar Hamed Abd [1,*], Mohammed Fadhil Mahdi [2], Luke K. Topham [3], Wasiq Khan [3], Sam Ansari [4,5] and Abir Hussain [4]

[1] College of Computer Science and Information Technology, University of Anbar, Ramadi 31001, Iraq
[2] Computer Technology Engineering Department, Al-Mansour University College, Baghdad 10068, Iraq; mohammed.fadhil@muc.edu.iq
[3] School of Computer Science and Mathematics, Liverpool John Moores University, Liverpool L3 5AH, UK; l.k.topham@ljmu.ac.uk (L.K.T.); w.khan@ljmu.ac.uk (W.K.)
[4] Department of Electrical Engineering, College of Engineering, University of Sharjah, Sharjah 27272, United Arab Emirates; sam@sharjah.ac.ae (S.A.); abir.hussain@sharjah.ac.ae (A.H.)
[5] Research Institute of Sciences and Engineering, University of Sharjah, Sharjah 27272, United Arab Emirates
* Correspondence: dhafar.hamed@uoanbar.edu.iq

Abstract

In the digital era, the rapid proliferation of fake news poses critical challenges to information credibility and public trust, particularly in Arabic news ecosystems where linguistic complexity and limited annotated resources exacerbate detection difficulties. This study proposes a framework for Arabic fake news detection based on a weighted-distance ensemble learning method (WDELM). The WDELM framework combines posterior-probability estimates from six heterogeneous base classifiers: extreme gradient boosting (XGBoost), LightGBM (LGBM), random forest (RF), adaptive boosting (AdaBoost), gradient boosting (GB), and logistic regression (LR). The classifier outputs are integrated through a distance-aware adaptive weighting strategy based on cosine distance in the prediction space. Unlike conventional ensemble techniques, the proposed framework employs normalised distance-aware adaptive weighting to adapt classifier contributions while preserving the probabilistic interpretation of the final ensemble output. Experiments were conducted on an Arabic fake news dataset comprising 2538 manually annotated instances. The model was evaluated using multiple performance metrics, including precision, recall, F1-score, Cohen's kappa, ROC-AUC, and accuracy, together with explainability analyses. Using stratified ten-fold cross-validation, the proposed WDELM achieved a mean accuracy of $92.120 \pm 1.097\%$ and a mean ROC-AUC of $97.125 \pm 0.686\%$. Analysis of the concatenated predictions generated across the ten outer-validation folds yielded an F1-score of 91.357% for the Fake class, an F1-score of 92.759% for the Real class, and a pooled macro-F1 score of 92.058%, indicating balanced classification performance across both classes. The results indicate that the proposed weighted-distance ensemble strategy provides improved empirical performance within the evaluated dataset and offers a transparent mechanism for combining heterogeneous classifiers. The framework is further assessed through statistical validation, error analysis, and explainability analysis, supporting its potential use as an auxiliary decision-support tool for Arabic fake news screening rather than as a fully automated replacement for professional fact-checking.

Keywords: machine learning; text classification; fake news detection; weighted distance ensemble learning (WDELM); Arabic natural language processing

Academic Editor: George Angelos Papadopoulos

Received: 26 June 2026
Revised: 5 August 2026
Accepted: 6 August 2026
Published: 14 August 2026

1. Introduction

The digitalisation of communication is no longer a phenomenon confined to Western countries. Social media has become almost ubiquitous in the Arab region, with approximately 79% of people in the Middle East using social media or direct messaging at least once a day [1]. Consequently, the way individuals access and evaluate news has undergone a fundamental transformation. An estimated 66% of users in the region now rely on social media as a primary source of news consumption [2]. Platforms such as WhatsApp, Facebook Messenger, Snapchat, and similar messaging applications have become dominant channels for information dissemination, particularly among younger populations [2,3]. In these environments, users increasingly rely on peer-to-peer sharing rather than institutional journalism, leading to a highly decentralised information ecosystem.

While this shift improves accessibility and speed of information dissemination, it also introduces a critical challenge: the rapid and uncontrolled spread of misinformation. Fake news in social media environments often propagates like an infectious process, amplifying fear, misinformation, and social instability [4,5]. The absence of systematic verification mechanisms among users further exacerbates this issue. As a result, misinformation has become a major concern for governments, media organisations, and researchers alike. Although educational and policy-driven interventions exist, they remain limited in effectiveness due to their reliance on individual user judgement and behavioural compliance.

With the rapid advancement of artificial intelligence, machine learning (ML)-based fake news detection has emerged as a promising computational solution [6]. A large body of research has investigated supervised learning models for detecting misinformation in multiple languages, particularly English. However, Arabic fake news detection remains comparatively underexplored due to linguistic complexity, limited labelled datasets, and morphological richness [7,8]. This gap is particularly critical given the socio-political instability in several Arabic-speaking regions, where misinformation can spread rapidly and have serious societal consequences. Therefore, robust automated approaches tailored specifically to Arabic textual characteristics are urgently required.

Recent studies have attempted to address this challenge by employing ensemble learning techniques, which can improve predictive performance and stability by combining complementary classifiers [9,10]. Ensemble methods such as bagging, boosting, and voting-based classifiers have been widely adopted for fake news detection, including on Arabic datasets [11,12]. These approaches generally demonstrate improved stability and accuracy by combining multiple base learners [13]. However, a critical limitation in most existing Arabic fake news ensemble models is the reliance on static, uniform combination strategies, such as majority voting or simple averaging, in which each classifier contributes equally to the final decision regardless of its reliability or contextual performance [14].

This assumption of equal contribution is theoretically weak and empirically suboptimal, especially in heterogeneous model settings where classifiers exhibit significantly different error distributions [15,16]. In addition, most existing ensemble approaches in Arabic fake news detection do not explicitly model inter-classifier relationships, such as prediction agreement, disagreement, or structural similarity between model outputs. As a result, ensemble decisions are often driven by aggregation rather than informed weighting, which can allow weak or unstable classifiers to influence final predictions negatively.

Another limitation of prior work is the insufficient quantitative exploitation of information about model diversity. Although diversity is a key factor in ensemble learning performance, most Arabic fake news detection studies treat it implicitly rather than integrating it directly into the weighting mechanism [17]. Few approaches attempt to measure

similarity or disagreement between classifiers in the prediction space, and even fewer incorporate such measures into distance-aware adaptive weighting strategies.

To address these limitations, this paper proposes a novel weighted distance ensemble learning method (WDELM) designed specifically for Arabic fake news detection. Unlike traditional ensemble approaches, WDELM introduces a distance-aware adaptive weighting mechanism that explicitly models the relationships between classifier predictions. In particular, the method utilises cosine distance to quantify the degree of agreement and disagreement among base learners in the prediction space.

The core intuition behind WDELM is that classifiers exhibiting higher consistency with the collective predictive behaviour of the ensemble should be assigned greater influence, while classifiers with highly divergent or unstable predictions should contribute less to the final decision. This formulation allows the ensemble to move beyond naive aggregation toward a distance-aware adaptive weighting framework, where inter-model similarity directly informs decision fusion.

In addition, the proposed approach operates on classifier prediction outputs rather than internal model parameters, allowing it to be applied to heterogeneous base learners that provide posterior-probability estimates. Because the weighting process is performed in the prediction space, it does not require modification of the internal structure of the constituent classifiers. Its robustness under substantial noise, severe class imbalance, or external domain shift, however, requires further empirical evaluation.

The primary methodological contribution of this study is the proposed WDELM, which introduces a distance-aware adaptive weighting mechanism for probabilistic ensemble learning. WDELM estimates normalised classifier weights from pairwise cosine distances computed on validation (out-of-fold) posterior probability vectors, enabling distance-aware classifier fusion that explicitly exploits inter-classifier agreement and diversity. Unlike conventional ensemble methods that assign equal or fixed weights, the proposed approach dynamically estimates normalised classifier contributions while preserving the probabilistic interpretation of the ensemble prediction. To demonstrate and validate the proposed method, this study further develops a comprehensive Arabic fake news detection framework by integrating Arabic-specific text preprocessing, linguistic feature extraction, a manually annotated Arabic fake news dataset constructed from publicly available Twitter content, and extensive experimental evaluation against conventional ML, DL, and state-of-the-art Arabic transformer models. This integrated framework provides a consistent experimental setting for evaluating representative Arabic fake news detection methods while accounting for selected linguistic characteristics of Arabic text.

The proposed method is evaluated on the constructed Arabic dataset and compared with representative machine learning, deep learning, ensemble, and pretrained transformer baselines using multiple evaluation metrics, including F1-score, precision, recall, and area under the receiver operating characteristic curve (ROC-AUC). The proposed framework advances Arabic fake news detection by shifting from conventional aggregation-based ensembles toward a distance-aware adaptive weighting paradigm. The integration of linguistic feature engineering and distance-based ensemble learning provides a more principled and adaptive approach to misinformation detection in Arabic social media environments.

The remainder of this paper is organised as follows. Section 2 reviews the relevant literature. Section 3 describes the proposed methodology, while Section 4 presents and discusses the experimental results. Finally, Section 5 concludes this paper and outlines directions for future research.

2. Related Work

The detection of fake news and misinformation has attracted significant research attention over the past decade [4,18]. Early studies primarily focused on English-language datasets due to the availability of large-scale annotated corpora and mature natural language processing (NLP) resources. However, in recent years, increasing attention has been directed toward low-resource languages, particularly Arabic, due to the rapid spread of misinformation in social media ecosystems across the Arab region. Arabic fake news detection remains challenging due to its rich morphology, dialectal variation, and limited annotated datasets. Section 2 reviews prior work in Arabic fake news detection, with a particular focus on machine learning, deep learning, and ensemble learning approaches.

2.1. Machine Learning Approaches for Arabic Fake News Detection

Traditional ML techniques remain widely used in Arabic fake news detection due to their simplicity and effectiveness in structured feature spaces. Several studies have explored classical classifiers such as support vector machine (SVM), random forest (RF), naïve Bayes (NB), and logistic regression (LR), typically combined with term frequency–inverse document frequency (TF-IDF) or n-gram-based feature representations.

For instance, the authors in [19] applied multiple classification algorithms to Twitter-based datasets with extensive preprocessing, including tokenization, removal of non-Arabic characters, and TF-IDF feature extraction. Feature selection using analysis of variance (ANOVA) further improved model performance, with RF achieving competitive results. Similarly, the study in [20] evaluated SVM, decision tree, and multinomial NB for rumor detection, reporting performance variability across topical domains and highlighting the sensitivity of ML models to contextual differences.

Further improvements have been explored through contextual feature engineering. Studies [21,22] integrated natural language processing techniques such as part-of-speech (POS) tagging and domain-specific lexical construction. These approaches demonstrated that incorporating linguistic structure improves classification accuracy, with RF achieving 79% accuracy in [21] and neural network-based models reaching up to 99.9% in [22].

Data augmentation strategies have also been proposed to mitigate data imbalance and overfitting. In [23], word-embedding-based augmentation with AraVec vectors significantly improved classification performance, increasing accuracy from 65% to 92% with an SVC when multiple feature categories were combined. Although these studies demonstrate the effectiveness of traditional ML pipelines, they primarily rely on single-model learning paradigms, limiting robustness and generalisation capability in noisy environments such as social media.

Model performance can be improved through the extraction of relevant textual features. The TF-IDF technique combines the frequency of a term within a document with its inverse frequency across the document collection, thereby assigning greater importance to terms with higher discriminative value. Several previous studies have used TF-IDF as a feature extraction approach [24–26] to provide classification algorithms with features for pattern extraction.

2.2. Deep Learning and Transformer-Based Models

Deep learning approaches have been widely adopted due to their ability to automatically learn hierarchical feature representations. Convolutional neural networks (CNNs), recurrent neural networks (RNNs), long-short-term memory (LSTM), and BiLSTM architectures have been extensively investigated for Arabic fake news detection [27].

The authors in [28] proposed a CNN-based model incorporating preprocessing techniques such as tokenization, lemmatization using Farasa, and stop-word removal. The

model achieved its best performance with optimised dropout and architecture configurations. Similarly, [29] compared CNN, LSTM, BiLSTM, and hybrid architectures, showing that BiLSTM achieved competitive performance (74–84%), although classical Linear SVC occasionally outperformed deep learning models on the same datasets.

Recent advances in Arabic NLP have been driven by pretrained transformer-based language models that learn rich contextual representations from large-scale Arabic corpora. Models such as MARBERT [30], MARBERTv2 [31], CAMeLBERT [32], QARiB [33], ArabicBERT [34], and AraELECTRA [35] have demonstrated state-of-the-art performance across various Arabic text classification tasks, including sentiment analysis, offensive language detection, and fake news detection. For example, ref. [36] combined MARBERT with CNN layers and achieved approximately 96% accuracy, while ref. [37] reported that transformer-based models consistently outperform traditional DL architectures across multiple datasets. Despite their superior classification performance, these models typically require substantial computational resources and often provide limited interpretability, particularly when integrated into ensemble learning frameworks. To provide a comprehensive comparison with contemporary Arabic NLP approaches, this study evaluates the proposed WDELM against seven representative pretrained Arabic transformer models, namely MARBERT, MARBERTv2, CAMeLBERT, QARiB, ArabicBERT, AraELECTRA, and AraBERTv2, in addition to conventional machine learning and deep learning methods.

2.3. Ensemble Learning for Arabic Fake News Detection

Ensemble learning has become a dominant strategy for improving classification performance by combining multiple base learners. Traditional ensemble methods such as bagging, boosting, and voting have been widely applied in Arabic fake news detection due to their ability to reduce variance and improve generalisation.

The study reported in [38] proposed a voting-based ensemble comprising multiple classifiers, including NB, linear SVC, LR, RF, stochastic gradient descent, and passive-aggressive classifiers. Feature extraction was performed using TF-IDF and word embeddings, followed by Chi-square feature selection. The results showed that an SVC model trained on content, interaction, and user features extracted from augmented data achieved an accuracy of 92%, compared with 65% without data augmentation.

Table 1 summarises representative Arabic fake news detection studies from 2021 to 2026, highlighting their methodological characteristics and limitations. As illustrated in Table 1, previous Arabic fake news detection studies have demonstrated promising results using machine learning, deep learning, and ensemble learning techniques; however, most existing ensemble approaches rely on static aggregation strategies and do not explicitly model inter-classifier relationships. Figure 1 presents a taxonomy of existing Arabic fake news detection approaches and highlights the methodological gap this study addresses.

As depicted in Figure 1, existing approaches primarily focus on improving individual classifier performance or combining classifiers through static voting mechanisms. In contrast, the proposed WDELM introduces a distance-aware adaptive weighting mechanism based on cosine-distance analysis in the prediction space, allowing classifier contributions to be adjusted according to their agreement with the remaining ensemble members.

Most existing methods assume equal or fixed contribution from each base learner, regardless of their contextual reliability or agreement behaviour. Even in dynamic ensemble approaches, weighting is typically based on individual classifiers' accuracy rather than on the structural relationships among classifiers. Consequently, weak or unstable models may still significantly influence the final decision.

Table 1. Comparison of Arabic fake news detection methods (2021–2026) vs. proposed WDELM.

Ref.	Year	Model Type	Ensemble Strategy	Arabic-Specific Processing	Feature Representation	Handling of Model Diversity	Weighting Strategy	Key Limitation	Proposed WDELM Advantage
[37]	2021	SVM, NB, RF	Majority Voting	Basic preprocessing	TF-IDF	Not considered	Equal voting	Ignores classifier reliability	WDELM assigns adaptive weights based on prediction similarity
[21]	2022	DL + ML ensemble	Hard voting	Limited Arabic normalisation	Word embeddings	Not explicitly modelled	Uniform aggregation	No inter-model relationship modelling	WDELM uses cosine-distance-based inter-model relationships
[39]	2022	CNN + LSTM ensemble	Soft voting	Partial Arabic stemming	Word embeddings	Partial diversity consideration	Fixed averaging	Sensitive to weak classifiers	WDELM reduces influence of unstable models dynamically
[40]	2023	Hybrid ML ensemble	Boosting-based ensemble	Arabic preprocessing pipeline	TF-IDF + n-grams	Implicit diversity via boosting	Sequential weighting	Focus on training error, not prediction similarity	WDELM directly models prediction-space similarity
[36]	2023	Stacking ensemble (SVM, LR, RF)	Meta-classifier stacking	Arabic stopword removal	TF-IDF	Limited diversity modelling	Meta learner only	Requires training a second-level model, less interpretable	WDELM is model-agnostic and does not require meta-learning
[41,42]	2024	Transformer + ML fusion	Weighted voting	Context-aware Arabic embeddings	BERT-based features	Not explicitly quantified	Static learned weights	High computational cost, limited interpretability	WDELM is lightweight and interpretable
[43]	2025	Ensemble of deep classifiers	Dynamic voting	Arabic dialect handling limited	Contextual embeddings	Partially dynamic	Accuracy-based weighting	Weighting based only on performance, not inter-model agreement	WDELM uses pairwise cosine distance for agreement modelling
[44]	2026	Hybrid ensemble systems	Ensemble averaging	Improved Arabic preprocessing	Mixed features	Not explicitly modelled	Equal averaging	No explicit diversity integration	WDELM explicitly incorporates diversity via distance metrics
Proposed Model WDELM		ML ensemble (heterogeneous)	Distance-based weighted ensemble	Arabic-specific NLP pipeline + lexical resources	TF-IDF/lexical features	Explicitly modelled via prediction vectors	Distance-aware adaptive weighting	–	Models inter-classifier relationships in the prediction space, assigns lower contributions to more divergent prediction patterns, and does not require a second-level meta-learner.

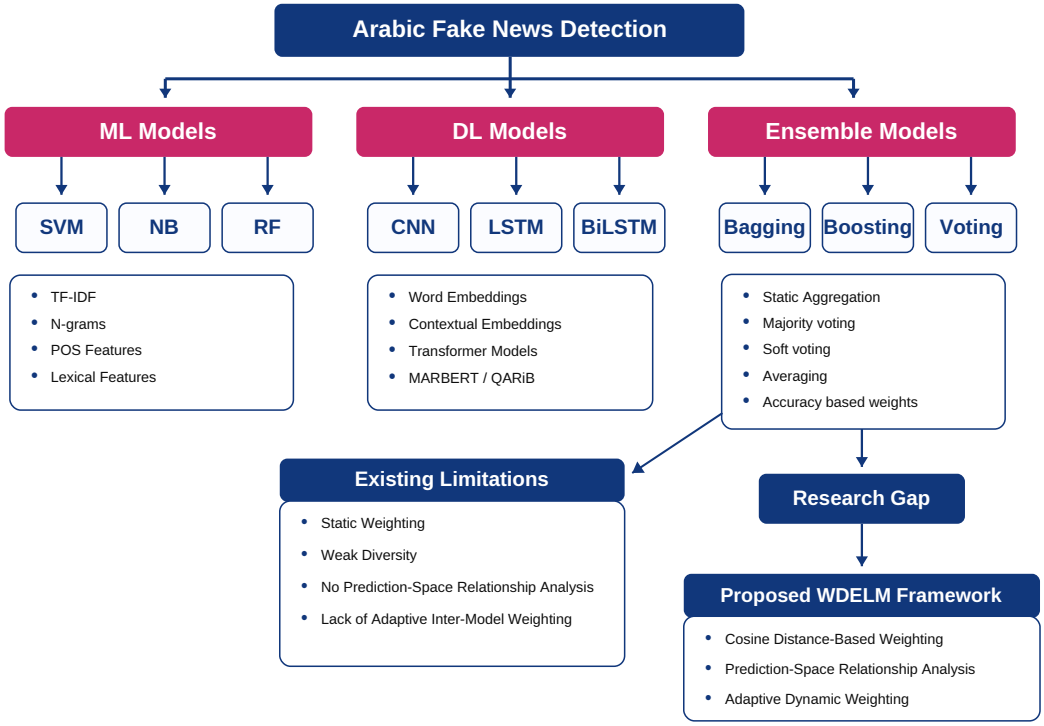

Figure 1. Taxonomy of Arabic fake news detection approaches, highlighting the methodological positioning of the proposed WDELM framework relative to existing ensemble strategies.

Furthermore, existing ensemble models rarely quantify classifier diversity in a principled mathematical form, particularly using distance-based measures in prediction space. Although diversity is recognised as a key factor in ensemble performance, it is often implicitly treated rather than explicitly incorporated into the decision fusion mechanism. There is a lack of ensemble frameworks that integrate explicit measures of inter-model similarity or disagreement into distance-aware adaptive weighting mechanisms.

To address the above limitations, this paper proposes a WDELM that fundamentally differs from prior ensemble approaches in Arabic fake news detection. Unlike traditional voting or averaging-based ensembles, WDELM introduces a distance-aware adaptive weighting mechanism that operates directly in the prediction space. Specifically, it computes pairwise cosine distances between classifier prediction vectors to quantify agreement and disagreement among base learners.

This allows the ensemble to move beyond static aggregation and instead adopt a distance-aware adaptive decision-fusion strategy, where:

- Classifiers with higher agreement with the ensemble consensus are assigned greater weights,
- Classifiers with inconsistent or divergent predictions are penalised,
- Model contributions are dynamically adjusted based on prediction-space structure rather than fixed rules.

WDELM operates on posterior-probability outputs and does not require a second-level meta-learner. It can therefore combine heterogeneous base classifiers that provide compatible probability estimates without modifying their internal architectures.

3. Methodology

Section 3 presents the methodological framework of the proposed study. The framework integrates data preprocessing, feature representation, model training, and a distance-aware adaptive ensemble-learning strategy. The following subsections describe the dataset

construction, preprocessing operations, feature representations, WDELM formulation, evaluation protocols, and explainability procedures.

Figure 2 illustrates the overall workflow, beginning with collecting data from Twitter to create a dataset of Arabic news via the Twitter API. The data then underwent five preprocessing steps: tokenization, stop-word removal, noise removal, normalisation, and root-stemming. After preprocessing, the resulting feature representations were evaluated under two explicitly separated protocols. The primary machine-learning, ablation, statistical, error-analysis, and explainability experiments employed a strictly fold-wise stratified ten-fold cross-validation procedure. For each outer fold, the corresponding outer-validation partition was isolated before any vocabulary construction, feature fitting, base-classifier training, posterior-probability generation, distance calculation, or WDELM weight estimation was performed.

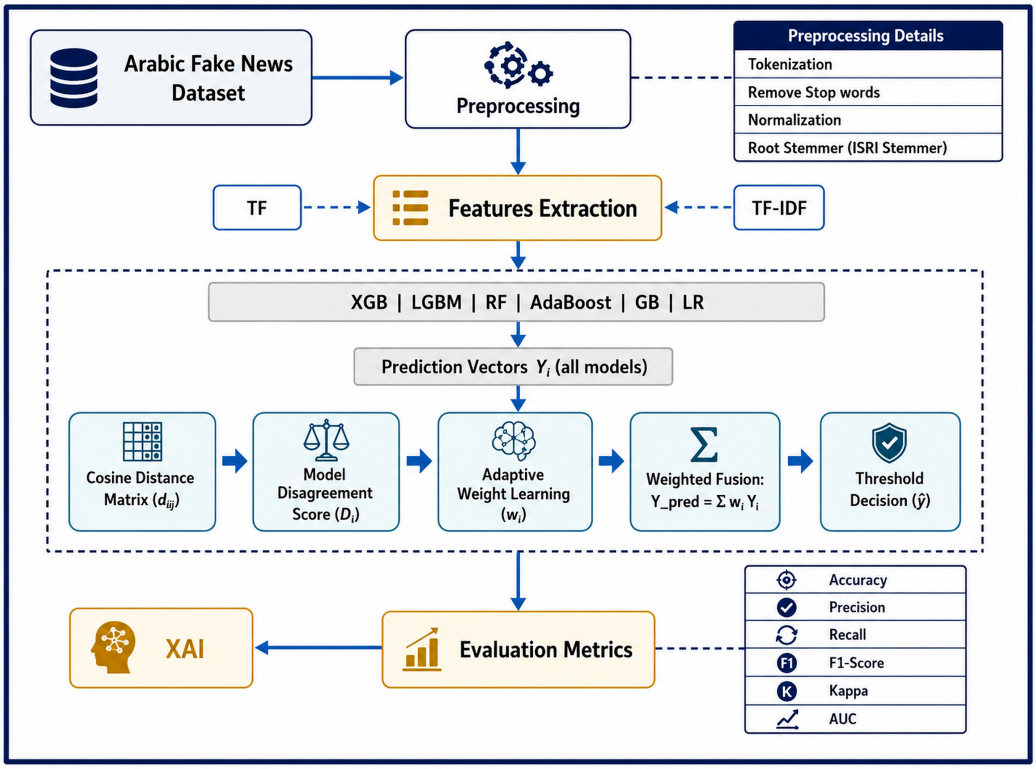

Figure 2. An illustration of the proposed model.

Within the corresponding outer-training partition, internal stratified cross-validation was used to generate out-of-fold posterior-probability vectors for the six base classifiers. These inner out-of-fold predictions were used exclusively to calculate the pairwise cosine distances, cumulative distance values, preliminary agreement scores, and normalised WDELM weights for that outer fold. The resulting weights were then frozen. The vocabulary, feature transformation, and base classifiers were subsequently fitted using the complete outer-training partition, and the frozen weights were applied to the posterior probabilities generated for the untouched outer-validation fold.

This procedure was applied separately to each of the ten outer folds. For every fold, all model fitting, inner out-of-fold probability generation, distance calculation, and WDELM weight estimation were conducted exclusively within the corresponding outer-training partition. The resulting weights were frozen before the associated outer-validation partition was evaluated. Only after all ten outer folds had been processed were their outer-validation predictions concatenated for aggregate confusion-matrix, ROC, and class-specific performance analyses. Although the outer-validation partitions were evaluated

separately, the fold-level performance estimates are not statistically independent because the corresponding outer-training partitions overlap. A separate stratified 70–30% train–test split was used exclusively for the comparison involving deep learning architectures and pretrained Arabic transformer models. The term frequency (TF) and term frequency-inverse document frequency (TF-IDF) techniques were employed to generate numerical representations of the textual content, which were subsequently used as inputs to the evaluated models. In the experimental study, ML and DL algorithms were employed as benchmark models to compare their performance with the proposed WDELM model, enabling a comprehensive evaluation of predictive accuracy and model effectiveness. Evaluation metrics, including precision, recall, F1-score, Cohen's kappa, AUC, and accuracy, were computed for each model.

In addition, Figure 2 illustrates the architecture of the proposed WDELM. Initially, the input dataset undergoes preprocessing and feature engineering before being distributed to the six heterogeneous base classifiers: XGBoost, LGBM, RF, AdaBoost, GB, and LR. These abbreviations are used consistently throughout the manuscript and its figure captions. For each outer fold r, the six base classifiers first generated inner out-of-fold posterior-probability vectors using only the corresponding outer-training partition. Pairwise cosine distances were calculated from these inner prediction vectors, and the resulting values were used to estimate a fold-specific normalised WDELM weight vector. The weights were frozen before the untouched outer-validation partition was processed.

The classifiers were then fitted on the complete outer-training partition and used to generate posterior probabilities for the corresponding outer-validation fold. The frozen fold-specific WDELM weights were applied to these probabilities to produce the final ensemble predictions for that fold. The complete pooled outer-validation prediction vector was formed only after this procedure had been applied separately to all ten outer folds. This concatenated prediction vector was used solely for aggregate performance reporting, confusion-matrix construction, ROC analysis, and class-specific evaluation.

For the separate deep-learning and pretrained-transformer comparison, the dataset is partitioned using a stratified 70–30% train–test split. In that protocol, all models are trained using only the training partition, and the held-out test partition is used exclusively for final performance evaluation. Results obtained under these two protocols are reported separately and are not treated as directly interchangeable.

3.1. Dataset Collection and Annotation Procedure

Detecting fake news in Arabic-language online environments remains challenging, primarily because of the limited availability of standardised benchmark datasets supported by transparent, reproducible, and verifiable annotation protocols. Since the performance and interpretability of fake-news detection models depend directly on the quality of the reference labels, particular attention was given to the construction of an evidence-based annotation framework.

The final dataset contains 2538 Arabic textual records collected from publicly accessible Twitter (X) posts between January 2024 and October 2025. The stated collection interval refers to the publication or retrieval dates of the posts rather than to the dates of the events discussed in their textual content. A keyword-based retrieval strategy was employed using the Arabic-language filter (`lang:ar`) and predefined Boolean AND/OR queries combining Arabic news-, claim-, and misinformation-related terms. The queries targeted news reporting, public claims, and socio-political discussions to increase topical coverage and reduce excessive concentration on a single discourse domain.

Only publicly accessible posts were considered, and no private or restricted content was accessed. During the original preprocessing procedure, retweets and direct reposts,

advertisements and spam, non-Arabic records, multimedia-only posts, deleted or inaccessible records, records containing insufficient textual content, and exact duplicate texts were excluded. Exact duplicates were identified after Arabic text normalisation using exact normalised-string matching.

The archived acquisition logs did not preserve the initial number of retrieved records or the separate number of records excluded at each filtering stage. These historical quantities cannot therefore be reconstructed reliably from the final corpus and were not estimated retrospectively. Similarly, source-category and dialect labels were not retained as structured per-record metadata. Consequently, reliable numerical source and dialect distributions cannot be reported without a separate validated annotation procedure.

The original preprocessing pipeline did not include a formally documented semantic near-duplicate screening stage. Accordingly, zero records were removed through formal near-duplicate detection during the original experiment. This does not establish that the final corpus contains no lightly edited reposts or paraphrases. To support future releases and independent replication, the protocol recommends semantic near-duplicate screening using word-level unigram–bigram and character-level 3–5-g TF–IDF representations, cosine similarity with a candidate threshold of 0.90, and manual verification before dataset partitioning.

Following preprocessing and annotation, the final high-confidence corpus comprised 2538 records, including 1158 records labelled as Fake and 1380 records labelled as Real. The complete data acquisition, preprocessing, annotation, and metadata-availability protocol is summarised in Table 2.

Table 2. Summary of the Twitter (X) data acquisition, preprocessing, annotation, and metadata-availability protocol.

Parameter	Description
Data source	Publicly accessible Twitter (X) posts
Collection period	January 2024–October 2025; dates refer to post publication or retrieval
Language	Arabic
Collection strategy	Predefined Boolean AND/OR queries combining Arabic news-, claim-, and misinformation-related terms
Search keywords	لقاح، انتخابات، علاج، سياسة، حرب، زلزال، خبر، عاجل، تصريح، إعلان، حقيقة، إشاعة، مزيف، كاذب، فيضانات مفبرك، فيروس،
Initial retrieval count	Not retained in the archived acquisition logs and not estimated retrospectively
Original exclusions	Retweets and direct reposts, advertisements and spam, non-Arabic records, multimedia-only posts, deleted or inaccessible records, records with insufficient textual content, and exact duplicate texts
Stage-specific exclusion counts	Not retained in the archived acquisition logs
Exact duplicate removal	Exact normalised-string matching after Arabic text normalisation and before annotation
Formal near-duplicate screening in the original experiment	Not applied; therefore, zero records were removed through a formal near-duplicate screening stage
Recommended near-duplicate protocol	Word-level unigram–bigram and character-level 3–5-g TF–IDF cosine similarity; candidate threshold ≥ 0.90, followed by manual review
Final dataset size	2538 high-confidence Arabic posts

Table 2. *Cont.*

Parameter	Description
Class distribution	Fake: 1158 records; Real: 1380 records
Annotation procedure	Independent manual assessment by three annotators, cross-source factual verification, and expert resolution of disagreements
Verification evidence	Official institutional statements, reputable Arabic news agencies, and recognised Arabic fact-checking sources
Platform-specific verification logs	Not retained at the individual-record level; no unsupported platform names or frequencies were assigned retrospectively
Source-category distribution	Not stored as structured per-record metadata and therefore unavailable
Dialect distribution	Not included in the original annotation schema and therefore unavailable
Data availability	Publicly available through Harvard Dataverse at https://doi.org/10.7910/DVN/A1JKYT

Note: The English translations of the Arabic search keywords are as follows: كاذب (false), مزيف (fake), إشاعة (rumor), حقيقة (truth), إعلان (announcement), تصريح (statement), عاجل (breaking), خبر (news), زلزال (earthquake), حرب (war), سياسة (politics), علاج (treatment), انتخابات (elections), لقاح (vaccine), فيروس (virus), مفبرك (fabricated), and فيضانات (floods).

Table 2 distinguishes between procedures implemented during the original dataset construction and procedures recommended for future releases. In particular, the original preprocessing pipeline included exact normalised-string duplicate removal but did not include a formally documented semantic near-duplicate screening stage. The initial retrieval count, stage-specific exclusion counts, source-category labels, dialect labels, and platform-specific verification logs were not retained and are therefore not reconstructed through unsupported estimates.

For future releases and independent replications, semantic near-duplicate screening should be completed before any training, validation, or testing partition is created. Let $\mathbf{v}_i$ and $\mathbf{v}_j$ denote TF–IDF representations of two normalised Arabic documents. Their cosine similarity is defined as

$$S_{ij} = \frac{\mathbf{v}_i^{\top}\mathbf{v}_j}{\|\mathbf{v}_i\|_2\|\mathbf{v}_j\|_2}. \quad (1)$$

Candidate pairs should be generated using both word-level unigram–bigram and character-level 3–5-g TF–IDF representations. A pair should be flagged when $S_{ij} \geq 0.90$ in either representation and subsequently reviewed manually to distinguish lightly edited reposts or paraphrases from independent reports describing the same event. When a pair is confirmed as a near duplicate, one representative record should be retained before dataset partitioning.

This formal semantic screening procedure was not implemented during the original experiment. Consequently, no records were removed through formal near-duplicate screening in the reported dataset. This statement should not be interpreted as evidence that the corpus contains no semantically similar or lightly edited records.

The keyword list was designed to cover multiple categories associated with misinformation dissemination rather than focusing on a single topic. Boolean combinations of topic-specific and misinformation-related keywords were used to retrieve content from different news domains. The retained records were processed using a consistent pipeline that included language filtering, exact duplicate removal, and quality-control procedures. However, because the original acquisition logs did not preserve the complete retrieval history or stage-specific exclusion counts, the exact composition of the reported corpus cannot be independently reconstructed from the retrieval protocol alone.

The dataset consists of two classes defined using explicit operational criteria. Fake text refers to content assessed as false, misleading, or unsupported through cross-verification against official institutional statements, reputable Arabic news agencies, and recognised Arabic fact-checking sources. Real text refers to content whose principal factual claims were consistent with independently verifiable authoritative sources. These operational definitions were based on factual consistency and cross-source verification rather than on linguistic style, political position, or the presence of individual keywords. The class distribution is presented in Table 3.

Table 3. Dataset class distribution of Arabic texts used in this study.

Class	Number of Instances
Fake	1158
Real	1380
Total	2538

To ensure annotation reliability, a structured multi-stage protocol was adopted. Each record was independently assessed by three annotators with experience in Arabic natural language processing and information verification. During the first stage, the annotators examined the principal factual claims in each post and compared them with official institutional statements, reputable Arabic news reports, and recognised Arabic fact-checking sources. During the second stage, the assigned labels were cross-verified against multiple independent sources whenever sufficient corroborating evidence was available. During the third stage, disagreements were reviewed by a domain expert, and records that could not be resolved with sufficient confidence were excluded from the final high-confidence corpus.

The original annotation records did not preserve a complete platform-specific verification log for every individual post. Consequently, the manuscript does not retrospectively attribute all annotations to particular fact-checking organisations or report unsupported platform-level frequencies. The verification evidence is therefore described according to source type, namely official institutional statements, reputable Arabic news agencies, and recognised Arabic fact-checking sources.

To standardize decision-making and minimise subjective bias, annotators followed a structured annotation guideline that defined labelling rules, verification procedures, and criteria for distinguishing among fake, real, and uncertain cases.

Ambiguous instances, including satire, opinion-based exaggeration, partially verified claims, or content lacking sufficient contextual evidence, were handled conservatively. Instances without sufficient verifiable evidence were excluded from the dataset. Cases with partial evidence were included only when confirmed by at least two independent credible sources. This conservative filtering strategy was adopted to reduce annotation noise and enhance epistemic reliability, ensuring that only high-confidence instances were retained.

To quantitatively evaluate annotation consistency, 20% of the collected news articles were randomly selected and independently annotated by all three annotators prior to the consensus stage. Each selected article therefore received three independent labels, which were used to assess inter-annotator reliability. Agreement was measured using Fleiss' Kappa, an agreement coefficient specifically designed for multiple annotators assigning categorical labels. The resulting agreement score was $\kappa = 0.84$, indicating substantial agreement according to the interpretation of Landis and Koch. Following the independent annotation phase, all disagreements were resolved through structured discussion and evidence-based verification until consensus was reached, and the agreed label was assigned as the final ground truth. The remaining 80% of the dataset was subsequently

annotated following the same annotation guidelines and consensus protocol. No additional samples from the finalized corpus were discarded during consensus resolution, because the annotators reached an agreed label for every retained news article.

Statistical characteristics of the dataset are summarised in Table 4. The analysis indicates that real texts tend to be longer and more structurally detailed, whereas fake texts exhibit relatively higher usage of numerical expressions. Despite these differences, both classes maintain comparable lexical diversity, indicating balanced linguistic coverage across the dataset.

Table 4. Statistical linguistic characteristics of fake and real text classes.

Feature	Fake Text	Real Text
Sentences	886	1260
Words	29,669	34,771
Characters	150,766	181,201
Numbers	568	473
Symbols	5251	7760
Punctuation Marks	1715	2040
Unique Words	7098	7147

It is important to emphasize that in fake news detection tasks, the reliability and operational definition of annotations directly determine the interpretability and validity of downstream performance results. In this study, annotation was treated as a controlled analytical process rather than a subjective labelling task. Each label was grounded in systematic external verification, and ambiguous cases were rigorously filtered or re-evaluated to minimise epistemic uncertainty. Accordingly, all reported experimental results should be interpreted as conditional on a high-confidence annotation protocol designed to reduce labelling noise, ensuring that model performance reflects genuine discriminative learning rather than inconsistencies in dataset construction.

3.2. Preprocessing

Preprocessing is a fundamental stage in NLP, aiming to transform raw, unstructured textual data into a clean and standardised format suitable for computational modelling. In the context of Arabic fake news detection, preprocessing plays a particularly critical role due to the language's rich morphology, extensive inflectional variations, and the presence of diacritics, elongation characters, and orthographic inconsistencies that may negatively affect feature representation and classification performance. In this study, a structured preprocessing pipeline was applied to all collected Arabic texts to ensure consistency and reduce linguistic noise prior to feature extraction and model training. The preprocessing workflow consisted of tokenization, stop-word removal, noise filtering, normalisation, and stemming.

a. Tokenization was performed to segment each text into a sequence of meaningful units (tokens), including words and punctuation marks. This step is essential for transforming continuous text into discrete elements that can be processed computationally. Given the characteristics of Arabic text, tokenization also helps address spacing inconsistencies and prepares the text for downstream feature extraction.
b. Stop-word removal was applied to eliminate high-frequency function words that carry limited discriminative semantic value for classification tasks. In fake news detection, content-bearing terms are more informative than grammatical words; therefore, removing stop words reduces feature space dimensionality and improves the model's discriminative capacity.

c. Noise removal involved eliminating non-informative textual elements such as punctuation marks, special symbols, and extraneous characters. Regular expressions were used to systematically identify and remove these elements based on predefined filtering rules. This step reduces structural noise that may otherwise distort statistical text representations.
d. Normalisation was applied to reduce orthographic variation in Arabic text and ensure lexical consistency. Specifically, this process included the removal of diacritics (Harakat), the unification of character variants (e.g., different forms of Alef and Ya), and the removal of Tatweel (_). Normalisation is particularly important in Arabic NLP, as it reduces sparsity caused by multiple surface forms of the same lexical item.
e. Stemming was performed using the Information Science Research Institute (ISRI) Arabic Stemmer, developed by the University of Nevada, Las Vegas [45]. The ISRI stemmer reduces words to their root forms without relying on a predefined dictionary, enabling generalisation across morphological variations. This is particularly beneficial in Arabic, where words derived from the same root may appear in multiple inflected forms.

The preprocessing pipeline was implemented using Python-based NLP libraries (e.g., NLTK and regular expression utilities), ensuring reproducibility of all transformations. The combination of normalisation and stemming helps mitigate morphological complexity, thereby improving the robustness of feature representation for downstream classification tasks in fake news detection.

3.3. Feature Extraction

Feature extraction is a fundamental stage in NLP [46], where raw textual data is transformed into structured numerical representations that ML models can process [47]. In Arabic fake news detection, this stage is particularly important due to the language's rich morphology, orthographic variability, and high lexical sparsity [48]. Effective feature extraction should therefore preserve discriminative textual information while reducing irrelevant variability and noise [49].

In this study, TF and TF-IDF were investigated because they provide computationally efficient, interpretable textual features while effectively capturing lexical patterns associated with deceptive and authentic content [50,51]. The term-frequency representation measures the relative occurrence of a term within a document. Let w_k denote the k-th term in the vocabulary and d_i denote the i-th document. The term frequency of w_k in d_i is defined as

$$\mathrm{TF}(w_k, d_i) = \frac{f_{k,i}}{\sum_{t=1}^{V} f_{t,i}}, \tag{2}$$

where $f_{k,i}$ denotes the number of occurrences of term w_k in document d_i, V is the vocabulary size, and $\sum_{t=1}^{V} f_{t,i}$ represents the total number of term occurrences in document d_i. Although TF captures the local importance of a term within a document, frequently occurring terms that appear in many documents may provide limited discriminative information.

To address this limitation, the inverse document frequency assigns greater importance to terms that occur in fewer documents. It is defined as

$$\mathrm{IDF}(w_k) = \log\left(\frac{N}{\mathrm{df}(w_k)}\right), \tag{3}$$

where N denotes the total number of documents in the corpus and $\mathrm{df}(w_k)$ denotes the number of documents containing term w_k.

The final TF-IDF weight of term w_k in document d_i is computed as

$$\text{TF-IDF}(w_k, d_i) = \text{TF}(w_k, d_i) \times \text{IDF}(w_k). \quad (4)$$

TF-IDF reduces the influence of terms that occur frequently across the corpus while emphasizing terms with greater document-specific discriminative value. Both TF and TF-IDF representations were generated after the completion of all preprocessing operations, and the resulting feature vectors were subsequently used as inputs to the evaluated classification models.

In this study, TF- and TF-IDF-based representations were adopted for several methodological reasons. First, the proposed framework prioritizes interpretability and computational efficiency, both of which are essential in misinformation detection tasks [30]. Unlike deep contextual embeddings, TF and TF-IDF provide transparent feature-weighting mechanisms that enable direct interpretation of influential textual terms. Second, transformer-based models introduce substantially higher computational complexity and require significantly larger training datasets to achieve effective generalisation [52]. Given the moderate size of the dataset used in this study (2538 instances), traditional statistical representations provide a computationally efficient and interpretable alternative for the present experimental setting. Finally, TF and TF-IDF remain strong and widely adopted baselines in Arabic NLP and text classification research, particularly for morphologically rich languages and medium-scale datasets. Their integration with Arabic-specific preprocessing techniques enables robust lexical representation while maintaining methodological simplicity and reproducibility.

Although both TF and TF–IDF were evaluated, the final representation was selected empirically rather than a priori. The TF-based configuration achieved the strongest overall performance under the common cross-validation protocol, including the highest accuracy, F1-score, Cohen's Kappa, and ROC-AUC. Therefore, TF was selected as the primary feature representation for the subsequent ablation, error, explainability, comparative, and statistical analyses. This selection is specific to the evaluated corpus and does not imply that TF is universally superior to TF–IDF for Arabic text classification.

3.4. Weighted Distance Ensemble Learning Method (WDELM)

Ensemble learning combines multiple ML classifiers to improve predictive accuracy, robustness, and generalisation performance. The underlying principle of ensemble learning is that different classifiers capture distinct patterns within the data; therefore, aggregating their predictions can reduce individual model bias and variance while improving overall classification reliability.

In this study, WDELM combines multiple classifiers using a distance-aware adaptive weighting mechanism. Unlike traditional voting strategies that assign equal importance to all classifiers, the proposed approach dynamically estimates model weights based on the similarity of their posterior probability outputs. Models that produce predictions more consistent with the ensemble consensus receive larger contribution weights in the final decision-making process.

Pairwise inter-model cosine similarity is subsequently computed to quantify prediction consistency among classifiers. The similarity values are transformed into cosine-distance measurements to estimate the level of classifier disagreement. A cumulative disagreement score is then calculated for each model and converted into adaptive importance weights using the proposed distance-aware adaptive weighting strategy. Models exhibiting lower disagreement with the ensemble consensus receive higher contribution weights during aggregation. Finally, a weighted fusion strategy is applied to combine the base-learner outputs into a unified probabilistic prediction under the adopted evaluation protocol.

Let the labelled dataset be defined as Equation (5).

$$D = \{(x_i, y_i) \mid i = 1, 2, \ldots, n\}, \tag{5}$$

where $x_i \in \mathbb{R}^d$ denotes the feature vector of sample i, $y_i \in \{0, 1\}$ denotes its corresponding binary label, and n is the total number of samples. The goal is to learn a classification function as shown in Equation (6).

$$f : \mathbb{R}^d \to \{0, 1\}, \tag{6}$$

The function f maps each input feature vector to a predicted binary class label. Let the ensemble consist of m base classifiers, as shown in (7).

$$\mathcal{M} = \{m_1, m_2, \ldots, m_m\}. \tag{7}$$

In this study, the ensemble comprises six heterogeneous base classifiers: XGBoost, LGBM, RF, AdaBoost, GB, and LR. Each classifier is trained independently on the training dataset. The proposed ensemble follows a parallel learning framework, in which all base classifiers are trained independently and simultaneously on the same training dataset D. Each model receives the same input feature space without any sequential dependencies or information sharing during training, which ensures diversity among learners and enhances the robustness of the ensemble. Formally, each classifier is defined as a mapping function $m_i : \mathbb{R}^d \to [0, 1]$, $\forall\, i = 1, 2, \ldots, m$, where $m_i(x)$ denotes the posterior probability estimated by the i^{th} classifier for sample x.

Under the primary stratified ten-fold cross-validation protocol, the WDELM weights were estimated separately within each outer fold. Let $\mathcal{D}^{(r)}_{\text{train}}$ and $\mathcal{D}^{(r)}_{\text{val}}$ denote the outer-training and outer-validation partitions for fold r, respectively. The outer-validation partition was not used during vocabulary construction, feature fitting, base-classifier training, posterior-probability generation for weight estimation, distance calculation, or weight estimation.

Within $\mathcal{D}^{(r)}_{\text{train}}$, internal stratified cross-validation was used to generate an inner out-of-fold posterior-probability vector for each classifier. For classifier m_i, this vector is denoted by $\mathbf{P}^{(r)}_{i,\text{inner}}$. The fold-specific pairwise cosine distances, cumulative distance values, preliminary agreement scores, and normalised weights were calculated exclusively from these inner out-of-fold vectors.

After the fold-specific weights had been estimated, they were frozen. The feature representation and the six base classifiers were then fitted using the complete outer-training partition, and posterior probabilities were generated for the untouched outer-validation partition. The frozen fold-specific weights were subsequently applied to those probabilities to obtain the WDELM predictions for outer fold r. No single global weight vector was estimated from the complete pooled outer-validation predictions.

For a given outer fold r, let n_r denote the number of observations in $\mathcal{D}^{(r)}_{\text{train}}$. The inner out-of-fold posterior-probability vector generated by classifier m_i within that outer-training partition is defined as

$$\mathbf{P}^{(r)}_{i,\text{inner}} = \left[p^{(r)}_{i1}, p^{(r)}_{i2}, \ldots, p^{(r)}_{in_r}\right], \tag{8}$$

where $p^{(r)}_{ik} \in [0, 1]$ denotes the posterior probability assigned by classifier m_i to the k-th observation in the inner out-of-fold prediction sequence for outer fold r. These vectors contain predictions only for observations in $\mathcal{D}^{(r)}_{\text{train}}$ and are used exclusively for estimating the WDELM weights for that outer fold.

The fold-specific cosine distance between classifiers m_i and m_j is computed as

$$d_{ij}^{(r)} = 1 - \frac{\sum_{k=1}^{n_r} p_{ik}^{(r)} p_{jk}^{(r)}}{\sqrt{\sum_{k=1}^{n_r} \left(p_{ik}^{(r)}\right)^2} \sqrt{\sum_{k=1}^{n_r} \left(p_{jk}^{(r)}\right)^2}}, \tag{9}$$

where smaller values indicate stronger agreement and larger values indicate greater disagreement between the two classifiers within the current outer-training partition. The cumulative disagreement of classifier m_i in outer fold r is

$$D_i^{(r)} = \sum_{\substack{j=1 \\ j \neq i}}^{m} d_{ij}^{(r)}, \qquad i = 1, 2, \ldots, m. \tag{10}$$

A preliminary fold-specific agreement score is then calculated as

$$\widetilde{w}_i^{(r)} = 1 - \frac{D_i^{(r)}}{\sum_{j=1}^{m} D_j^{(r)}}, \tag{11}$$

and normalised according to

$$w_i^{(r)} = \frac{\widetilde{w}_i^{(r)}}{\sum_{j=1}^{m} \widetilde{w}_j^{(r)}}, \tag{12}$$

which guarantees

$$\sum_{i=1}^{m} w_i^{(r)} = 1. \tag{13}$$

After the weights have been estimated, the vocabulary, feature transformation, and base classifiers are refitted using the complete $\mathcal{D}_{\text{train}}^{(r)}$. Let $\mathbf{P}_{i,\text{val}}^{(r)}$ denote the posterior-probability vector generated by classifier m_i for the untouched outer-validation partition. The final WDELM posterior-probability vector for outer fold r is

$$\mathbf{P}_{\text{WDELM,val}}^{(r)} = \sum_{i=1}^{m} w_i^{(r)} \mathbf{P}_{i,\text{val}}^{(r)}. \tag{14}$$

The fold-specific weight vector $\mathbf{w}^{(r)} = [w_1^{(r)}, \ldots, w_m^{(r)}]$ is frozen before any posterior probability is generated for $\mathcal{D}_{\text{val}}^{(r)}$. Thus, no observation in the corresponding outer-validation partition contributes to feature fitting, classifier training, distance calculation, or weight estimation.

To clarify the exact role of normalisation, the ensemble output before final weight normalisation is defined as

$$\widetilde{\mathbf{P}}_{\text{WDELM,val}}^{(r)} = \sum_{i=1}^{m} \widetilde{w}_i^{(r)} \mathbf{P}_{i,\text{val}}^{(r)}, \tag{15}$$

where $\widetilde{w}_i^{(r)}$ denotes the preliminary agreement score assigned to classifier m_i in outer fold r. Based on the definition of the preliminary agreement score, the sum of these scores is obtained as

$$\sum_{i=1}^{m} \widetilde{w}_i^{(r)} = \sum_{i=1}^{m} \left(1 - \frac{D_i^{(r)}}{\sum_{j=1}^{m} D_j^{(r)}}\right) = m - \frac{\sum_{i=1}^{m} D_i^{(r)}}{\sum_{j=1}^{m} D_j^{(r)}} = m - 1. \tag{16}$$

Accordingly, the final normalised weight assigned to classifier m_i can be written as

$$w_i^{(r)} = \frac{\widetilde{w}_i^{(r)}}{m-1}. \tag{17}$$

Therefore, the relationship between the unnormalised ensemble output and the final normalised WDELM output is

$$\widetilde{\mathbf{P}}_{\mathrm{WDELM,val}}^{(r)} = (m-1)\mathbf{P}_{\mathrm{WDELM,val}}^{(r)}. \tag{18}$$

Equation (18) shows that removing the final normalisation step only rescales the ensemble output by a positive constant and does not change the ranking of the samples. Consequently, the ROC-AUC values of the normalised and unnormalised outputs are identical. Threshold-dependent binary predictions are also equivalent when the decision threshold applied to the unnormalised output is changed from 0.5 to $0.5(m-1)$. Because the present ensemble contains six base classifiers, the equivalent threshold for the unnormalised output is 2.5. Therefore, normalisation is required to constrain the ensemble output to the interval $[0, 1]$, preserve its probabilistic interpretation, and permit the use of the conventional decision threshold of 0.5. It should not be interpreted as an independent source of improved discriminative performance.

Let $q_r = |\mathcal{D}_{\mathrm{val}}^{(r)}|$ denote the size of the outer-validation partition. Since each component of $\mathbf{P}_{i,\mathrm{val}}^{(r)}$ lies in $[0, 1]$ and the fold-specific weights satisfy Equation (13), the WDELM output for outer fold r is bounded by

$$\mathbf{P}_{\mathrm{WDELM,val}}^{(r)} \in [0,1]^{q_r}. \tag{19}$$

A fixed threshold of 0.5, specified before evaluation, was applied to every component of the normalised outer-validation probability vector:

$$\widehat{y}_k^{(r)} = \begin{cases} 1, & P_{\mathrm{WDELM,val},k}^{(r)} \geq 0.5, \\ 0, & P_{\mathrm{WDELM,val},k}^{(r)} < 0.5, \end{cases} \qquad k = 1, \ldots, q_r. \tag{20}$$

The threshold was not selected or optimised using the corresponding outer-validation partition.

For Variant A5, the unnormalised fold-specific output satisfies $\widetilde{\mathbf{P}}_{\mathrm{WDELM,val}}^{(r)} = (m-1)\mathbf{P}_{\mathrm{WDELM,val}}^{(r)}$. Hence, the equivalent threshold is $0.5(m-1)$, which equals 2.5 for the six-classifier ensemble. This positive rescaling preserves the ranking of the outer-validation observations and therefore leaves ROC-AUC unchanged.

The software implementation of the complete WDELM follows Equations (11)–(20). Specifically, the preliminary agreement scores are computed from the cumulative cosine-distance values using Equation (11), normalised according to Equation (12)–(13), and sub-

sequently applied to the posterior probability vectors of the base classifiers through Equation (14). The normalised weights form a convex combination, and the resulting ensemble output consequently satisfies Equation (19). Thus, the complete normalised WDELM retains a valid probabilistic interpretation and supports the conventional decision threshold of 0.5. Algorithm 1 summarises the proposed WDELM framework for fake news classification.

Algorithm 1: Strictly Fold-Wise Evaluation of the Proposed WDELM

Input: Labelled dataset $\mathcal{D}$, base-classifier set $\mathcal{M} = \{m_1, m_2, \ldots, m_m\}$, number of outer folds $K_{\text{out}} = 10$, number of inner folds K_{in} used in the implementation, and fixed random seed

Output: Pooled outer-validation posterior probabilities and predicted class labels

Generate fixed stratified outer-fold assignments;

for $r = 1$ **to** K_{out} **do**

- Define $\mathcal{D}_{\text{train}}^{(r)}$ and $\mathcal{D}_{\text{val}}^{(r)}$;
- Isolate $\mathcal{D}_{\text{val}}^{(r)}$ from vocabulary construction, feature fitting, model training, probability generation for weight estimation, distance calculation, and weight estimation;
- Generate stratified inner folds using only $\mathcal{D}_{\text{train}}^{(r)}$;
- **foreach** *inner split* **do**
  - Fit the vocabulary and feature transformation using only the inner-training subset;
  - Transform the inner-training and inner-validation subsets using the fitted transformation;
  - **foreach** $m_i \in \mathcal{M}$ **do**
    - Train m_i on the transformed inner-training subset;
    - Generate posterior probabilities for the inner-validation subset;
    - Store the probabilities in $\mathbf{P}_{i,\text{inner}}^{(r)}$;
- Compute the fold-specific pairwise cosine distances $d_{ij}^{(r)}$ from the inner out-of-fold probability vectors;
- Compute $D_i^{(r)}$, $\tilde{w}_i^{(r)}$, and the normalised weights $w_i^{(r)}$ using (10)–(13);
- Freeze the fold-specific weight vector $\mathbf{w}^{(r)}$;
- Fit the vocabulary and feature transformation using the complete $\mathcal{D}_{\text{train}}^{(r)}$;
- Retrain all base classifiers using the complete transformed outer-training partition;
- Generate $\mathbf{P}_{i,\text{val}}^{(r)}$ for the untouched outer-validation partition;
- Compute $\mathbf{P}_{\text{WDELM,val}}^{(r)} = \sum_{i=1}^{m} w_i^{(r)} \mathbf{P}_{i,\text{val}}^{(r)}$;
- Store the outer-validation posterior probabilities and predicted labels;

Concatenate the predictions obtained from all ten outer-validation folds in their original sample order;

Do not recompute the WDELM weights from the concatenated outer-validation predictions;

Compute the aggregate confusion matrix, ROC curve, class-specific metrics, and pooled performance values only after all outer folds have been completed;

return *Pooled outer-validation probabilities and predicted class labels*

The computational complexity of the proposed WDELM is derived from an analysis of its main computational components. Let m denote the number of base models and n denote the number of input samples. Initially, each base model generates predictions for all samples, incurring a computational cost of $\mathcal{O}(mn)$. Subsequently, the pairwise cosine similarity (or distance) between base models is computed to capture inter-model relationships, where all m models are compared in a pairwise manner. Since each comparison operates on prediction vectors of length n, this step incurs a cost of $\mathcal{O}(m^2n)$. The model weights are then computed via a normalisation step across the m models, incurring an $\mathcal{O}(m)$ cost. Finally, the ensemble output is obtained through a weighted aggregation of predictions across all models and samples, which requires $\mathcal{O}(mn)$ operations. Therefore, the additional ensemble-fusion complexity is dominated by the pairwise distance computation, yielding a complexity of $\mathcal{O}(m^2n)$. When the number of base learners, m, is fixed, this component grows linearly with the number of prediction instances, n. However, the present study did not conduct empirical large-scale scalability, memory-consumption, or deployment-latency experiments; therefore, no claim of operational scalability beyond the evaluated dataset is made.

To demonstrate the operational mechanism of the proposed WDELM, a small-scale illustrative case study is presented in Figure 3. The objective is to clarify the computational procedure of the model, including prediction generation, inter-model similarity computation, adaptive weight estimation, and final ensemble decision-making. This example is intended solely to clarify the computational procedure of WDELM and improve the reproducibility of its implementation; it is not used for performance evaluation.

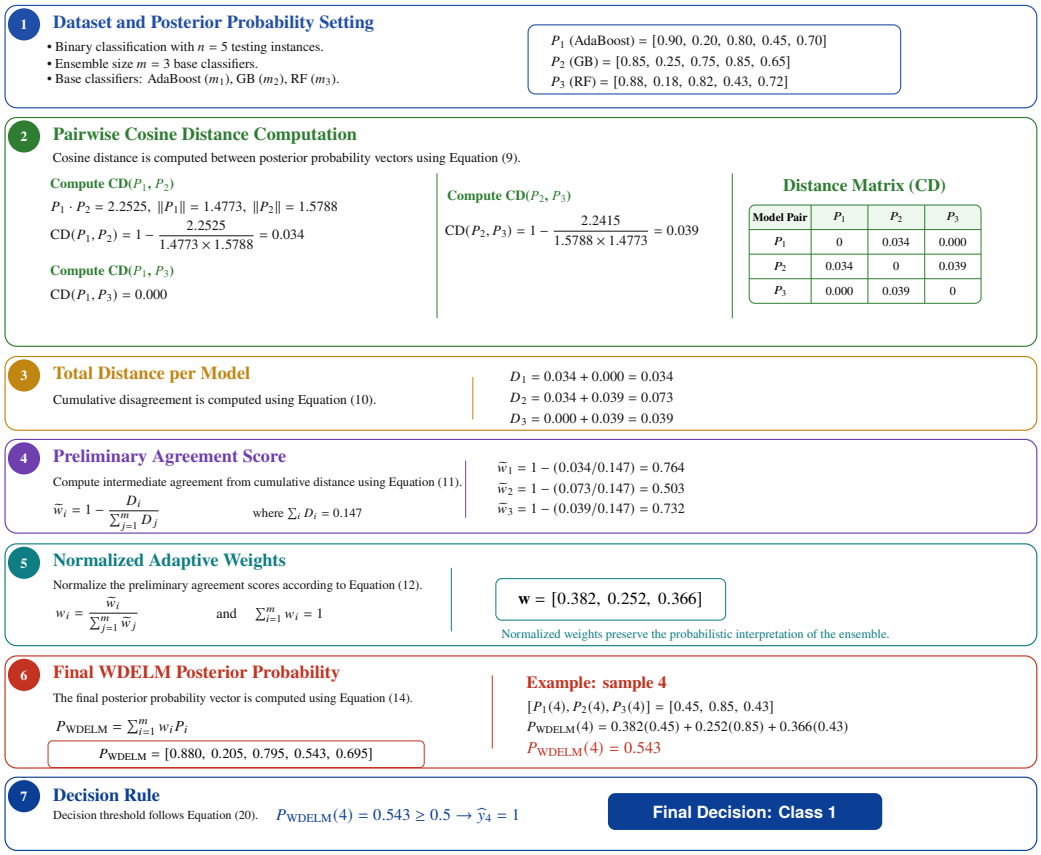

Figure 3. Comprehensive infographic of the proposed WDELM methodology from data acquisition to final prediction and model-output explanation.

3.5. Evaluation Metrics

Precision, recall, F1-score, Cohen's Kappa, ROC-AUC, and accuracy were used to evaluate the classification performance of the examined models. The class encoding adopted in the implementation was

$$\text{Fake} = 0, \qquad \text{Real} = 1. \tag{21}$$

Accordingly, the binary precision, recall, and F1-score values originally computed for the fold-level comparison tables used `pos_label=1` and therefore represent performance for the Real class. To remove ambiguity, these quantities are referred to throughout the manuscript as Real-class precision, Real-class recall, and Real-class F1-score.

Because the principal application concerns the discrimination of both fake and real news, the evaluation additionally reports class-specific metrics for both classes and macro-averaged metrics. For class $c \in \{\text{Fake}, \text{Real}\}$, precision, recall, and F1-score are defined as

$$\text{Precision}_c = \frac{\text{TP}_c}{\text{TP}_c + \text{FP}_c}, \tag{22}$$

$$\text{Recall}_c = \frac{\text{TP}_c}{\text{TP}_c + \text{FN}_c}, \tag{23}$$

and

$$\text{F1}_c = 2 \times \frac{\text{Precision}_c \times \text{Recall}_c}{\text{Precision}_c + \text{Recall}_c}. \tag{24}$$

The macro-averaged value of a class-specific metric M is calculated by assigning equal importance to the two classes:

$$M_{\text{macro}} = \frac{1}{2}(M_{\text{Fake}} + M_{\text{Real}}). \tag{25}$$

Macro-F1 was used as the principal class-balanced summary metric in the interpretation because it gives equal weight to the Fake and Real classes and does not allow the performance of one class to dominate the overall assessment. Accuracy, Cohen's Kappa, and ROC-AUC were retained as complementary measures of overall correctness, agreement beyond chance, and ranking-based discrimination, respectively.

The fold-level values reported as mean ± standard deviation were computed separately within each of the ten stratified cross-validation folds. The class-wise and macro-averaged values were calculated once from the concatenated predictions of the ten separately evaluated outer-validation folds. Consequently, pooled outer-validation metrics and fold-averaged metrics represent different aggregation procedures and should not be expected to be numerically identical in every case.

To provide a descriptive summary of fold-to-fold variability, a two-sided 95% fold-based interval was calculated for each principal metric as

$$\text{FI}_{95\%} = \bar{x} \pm t_{0.975,K-1}\frac{s}{\sqrt{K}}, \tag{26}$$

where $\bar{x}$ denotes the mean metric value across the $K = 10$ observed outer folds, s denotes the corresponding sample standard deviation, and $t_{0.975,K-1}$ denotes the critical value of the Student's t distribution with $K - 1 = 9$ degrees of freedom. Because the cross-validation training sets overlap, the fold-level estimates are correlated rather than independent. Accordingly, these quantities are reported only as descriptive 95% fold-based intervals summarising variation among the ten observed fold values. They are not interpreted as conventional population-level confidence intervals or as corrected estimates of uncertainty across independent external datasets. More rigorous inferential uncertainty es-

timation would require repeated cross-validation, an appropriately corrected resampling procedure, or evaluation across independent external datasets.

3.6. Reproducibility Artifacts and Replication Package

To support direct verification of the proposed weighting procedure and the associated statistical analyses, the revised study is accompanied by a structured replication package. The package preserves the complete fold-specific experimental outputs required to reconstruct the WDELM and ablation results under the adopted strictly fold-wise stratified ten-fold cross-validation protocol. All stochastic operations use the fixed random seed 42, and the exact outer- and inner-fold assignments are archived with sample identifiers to permit direct verification of the data partitions.

For each outer fold r, the replication package reports the posterior probabilities generated by XGBoost, LGBM, RF, AdaBoost, GB, and LR for the corresponding validation observations. It also reports the pairwise cosine distances $d_{ij}^{(r)}$, cumulative distances $D_i^{(r)}$, preliminary agreement scores $\widetilde{w}_i^{(r)}$, and final normalised weights $w_i^{(r)}$. These records permit independent reconstruction of both the complete WDELM prediction and the A7 equal-weight soft-voting prediction from the same six fold-specific posterior-probability vectors. The package further contains the fold-specific performance metrics, paired fold-level differences, effect-size calculations, and executable scripts used for the complete WDELM, all ablation variants, and the Friedman, Wilcoxon–Holm, and Nemenyi analyses.

Table 5 summarises the contents and verification purpose of the archived materials. The accompanying `README` specifies the execution order, software requirements, file structure, column definitions, and commands required to reproduce the reported tables and figures.

Table 5. Contents of the supplementary replication package for independent verification of the WDELM evaluation.

Replication Artifact	Verification Purpose
Random seed and software configuration	Reports the fixed seed 42, package versions, and execution settings used for data partitioning, model fitting, probability generation, and statistical analysis.
Exact outer- and inner-fold assignments	Provides sample identifiers and fold membership for every outer and inner cross-validation partition.
Fold-specific base-model probabilities	Reports the six posterior probabilities, true labels, and sample identifiers for every outer-validation observation.
Distances, preliminary scores, and weights	Reports $d_{ij}^{(r)}$, $D_i^{(r)}$, $\widetilde{w}_i^{(r)}$, and $w_i^{(r)}$ for every classifier and outer fold.
Fold-specific performance metrics	Reports precision, recall, F1-score, Cohen's kappa, ROC-AUC, and accuracy for every model and fold.
Paired fold-level differences and effect sizes	Reports paired metric differences, signed ranks, Wilcoxon statistics, unadjusted and Holm-adjusted p-values, and pairwise effect sizes.
Complete WDELM and ablation code	Contains executable implementations of the complete WDELM, A1–A7, and all leave-one-classifier-out configurations.
Statistical-analysis code	Contains the scripts used for the Friedman tests, Kendall's W, Wilcoxon–Holm comparisons, pairwise effect sizes, and Nemenyi analysis.

Note: All fold-specific files preserve sample identifiers and fold membership. Therefore, the complete WDELM and A7 predictions can be reconstructed from the same base-classifier posterior probabilities without using information from the corresponding outer-validation partition during model fitting or weight estimation.

4. Results and Discussion

Section 4 presents a comprehensive evaluation of the proposed Arabic fake news detection framework. The analysis includes feature representation comparison, model performance, statistical validation, robustness analysis, ablation study, and explainability using XAI techniques. Unless otherwise stated, the results in Sections 4.1–4.7 were obtained using stratified ten-fold cross-validation. The separate 70–30% train–test protocol is used only in the deep-learning and pretrained-transformer comparative analysis.

4.1. Dataset Analysis and Experimental Setup

Table 6 presents the statistical analysis of two features using two metrics (mean and variance). The mean value of 0.006 indicates that, on average, the TF across the dataset is 0.006. This value suggests that the words are relatively rare in the dataset. The variance of 0.006 is relatively high relative to the mean, indicating considerable variability in TF values. Some terms might appear much more frequently than others. The mean value of 0.001 indicates that, on average, the TF-IDF scores are lower than the TF scores. This lower mean value is expected because TF-IDF considers TF while also down-weighting words that appear in many documents, reducing their overall weight. The reported TF-IDF variance rounds to 0.000 at the displayed precision, indicating substantially lower variability than that observed for TF. This suggests that the terms are either uniformly distributed or that the TF-IDF transformation has effectively normalised word importance across documents.

Table 6. Statistical values for the TF and TF-IDF features.

Feature	Mean	Variance
TF	0.006	0.006
TF-IDF	0.001	0.000

Figure 4 shows the difference between the TF and TF-IDF features using Principal Component Analysis (PCA), which provides additional insight into the structure and distribution of the text data. Furthermore, Figure 4 shows how these two text data representations differ in their distributions and clustering. TF-IDF generally provides a more nuanced representation by accounting for the importance of terms relative to the document dataset, resulting in clustering patterns that differ from those produced by TF. In TF, there is some overlap between the fake and real documents, but distinct clusters also emerge. The spread of the points indicates substantial variance in the term frequencies captured by PCA. Compared to the TF plot, the TF-IDF plot shows a more distinct separation between the fake and real documents, particularly along PC1. The points are more tightly clustered, suggesting that TF-IDF provides a more discriminative feature space.

The predictive behaviour of machine learning models is influenced by their hyperparameter settings. Therefore, the principal hyperparameters of the evaluated classifiers were configured before the comparative experiments. Table 7 summarises the parameter settings used for the base classifiers and the proposed WDELM ensemble. The same configurations were retained across the corresponding evaluation protocols to support consistent comparisons.

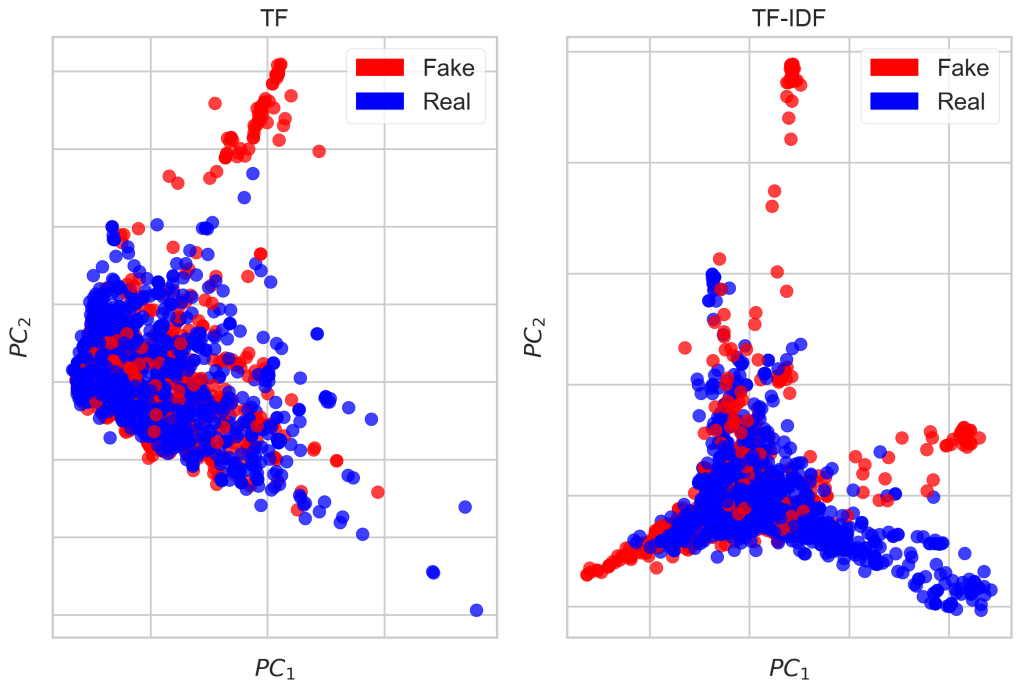

Figure 4. PCA of the TF and TF-IDF feature representations.

Table 7. Hyperparameter settings of the base classifiers used in the proposed WDELM framework.

Algorithm	Parameter
LR	$C = 1.0$, Solver = lbfgs, max_iter = 100
XGBoost	n_estimators = 100, learning_rate = 0.1, max_depth = 6, min_child_weight = 1, subsample = 1.0, colsample_bytree = 1.0
RF	n_estimators = 100, criterion = gini, min_samples_split = 2, min_samples_leaf = 1
GB	n_estimators = 100, learning_rate = 0.1, max_depth = 3, min_samples_split = 2, min_samples_leaf = 1
LGBM	n_estimators = 100, learning_rate = 0.1, num_leaves = 31, max_depth = -1, min_data_in_leaf = 20, bagging_fraction = 0.8, feature_fraction = 0.8
AdaBoost	n_estimators = 50, learning_rate = 0.1, algorithm = SAMME.R
WDELM	Ensemble of XGBoost, LGBM, RF, AdaBoost, GB, and LR

4.2. Performance of the WDELM Model

Table 8 compares the evaluated WDELM configurations using TF and TF–IDF representations under five-fold and ten-fold cross-validation. The TF-based configuration achieved the strongest overall results and was therefore selected as the primary configuration for the remaining analyses. The discussion emphasizes the F1-score as the principal balance between class-specific precision and recall, while accuracy, Cohen's Kappa, and ROC-AUC are reported as complementary performance indicators.

The five-fold and ten-fold cross-validation results show relatively consistent performance, with limited variability across the evaluated folds. These findings indicate stable predictive behaviour under the adopted internal validation protocols; however, they do not independently establish generalisation to external datasets or completely exclude the possibility of overfitting.

The low standard deviations observed across all evaluation metrics indicate that WDELM produces stable predictions over repeated cross-validation folds. However, descriptive statistics alone cannot determine whether the observed improvements are statisti-

cally significant compared with competing approaches. Therefore, formal non-parametric statistical tests are presented in Section 4.7. For further analysis, Figures 5 and 6 show comparisons between stratified five- and ten-fold cross-validation and the learning curve of the proposed model, respectively.

Table 8. Performance comparison of TF and TF-IDF feature representations under different cross-validation settings. Precision, recall, and F1-score are reported for the Real class using `pos_label = 1`.

Metrics (%) Mean ± Std	TF (CV = 5)	TF (CV = 10)	TF-IDF (CV = 5)	TF-IDF (CV = 10)
Real-class Precision	91.369 ± 1.076	92.748 ± 1.812	90.301 ± 1.913	91.471 ± 1.693
Real-class Recall	91.884 ± 0.746	92.826 ± 2.111	91.304 ± 0.760	91.957 ± 2.369
Real-class F1-score	91.620 ± 0.582	92.756 ± 1.019	90.786 ± 0.941	91.683 ± 1.233
Kappa	81.566 ± 1.351	84.117 ± 2.213	79.643 ± 2.279	81.727 ± 2.598
AUC	96.815 ± 0.339	97.125 ± 0.686	96.409 ± 0.378	96.772 ± 0.690
Accuracy	90.859 ± 0.664	92.120 ± 1.097	89.913 ± 1.113	90.938 ± 1.295

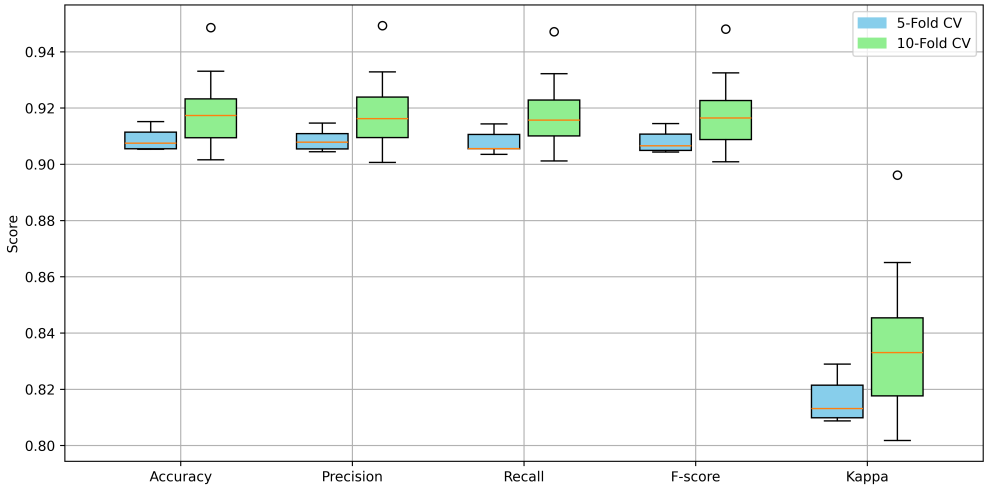

Figure 5. Performance comparison between stratified five- and ten-fold cross-validation using different evaluation metrics.

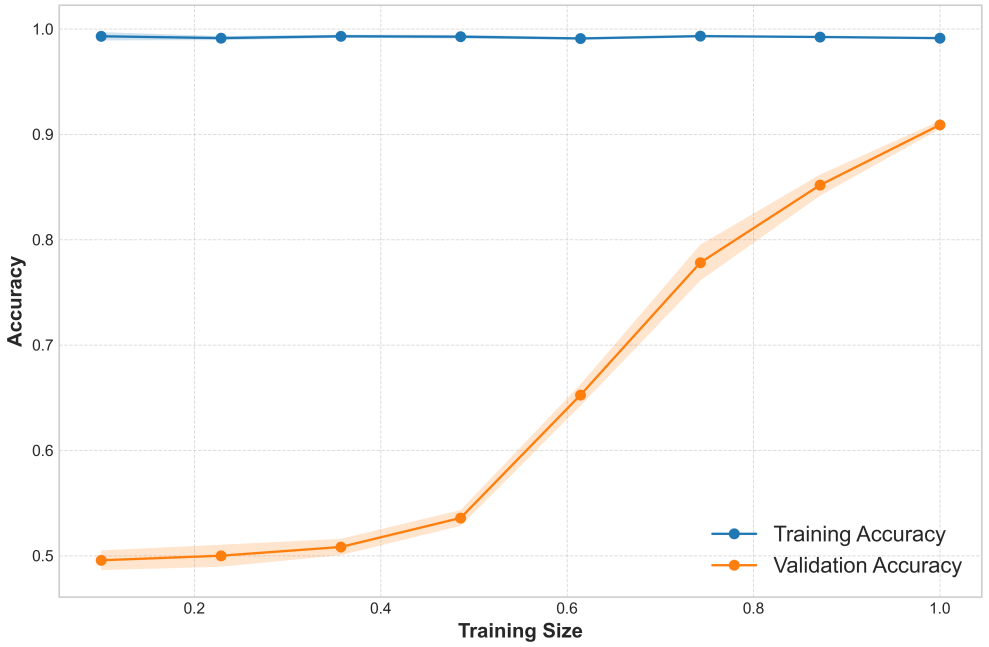

Figure 6. Learning curve analysis of the proposed model using different training sizes.

Figures 5 and 6 present the internal performance stability and learning behaviour of the proposed model under different validation strategies and training sizes. The five-fold and ten-fold cross-validation results show relatively high performance across accuracy, precision, recall, and F1-score, with the ten-fold setting producing slightly higher median values. Although Cohen's Kappa exhibited greater variability than the other metrics, the overall results remained comparatively stable across the evaluated folds. These findings describe internal cross-validation behaviour and should not be interpreted as evidence of external generalisation.

Figure 6 illustrates the learning curve analysis of the proposed approach. The training accuracy remained consistently high at approximately 99% across all training sizes, while the validation accuracy progressively increased as more training samples were introduced. Specifically, validation accuracy was close to 50% at the smallest training size, indicating that the limited subset did not provide sufficient coverage of the corpus's lexical variability and class-dependent patterns for reliable validation predictions. Although the constituent models could fit this small training subset, the learned decision boundaries were not yet sufficiently stable for unseen records. Validation accuracy progressively increased to approximately 91% as additional training samples provided broader and more representative coverage of the dataset. The learning-curve and cross-validation results indicate that performance became more stable as the training-set size increased. Nevertheless, these internal results do not directly establish reduced overfitting or generalisation to independent external datasets.

4.3. Ablation Study

In order to examine the contribution of the principal components of the proposed WDELM, the ablation study was conducted at two levels. First, the effects of inner out-of-fold posterior-probability estimation, distance measure, weighting strategy, normalisation, and fusion mechanism were examined. Second, each of the six base classifiers was individually excluded from the complete ensemble to evaluate the performance change associated with its removal. In addition, a controlled equal-weight soft-voting baseline, denoted by A7, was introduced to isolate the contribution of the proposed distance-aware weighting rule. A7 uses the same six classifiers, outer-fold assignments, posterior-probability predictions, fusion rule, and decision threshold as the complete WDELM, while assigning the fixed weight $w_i = 1/m = 1/6$ to each classifier. The exact definitions of the evaluated configurations are presented in Table 9, while their quantitative results are reported in Table 10.

The ablation results were obtained using stratified ten-fold cross-validation. Therefore, their values differ from those obtained under the separate 70–30% train–test protocol used for the comparison with the deep-learning and pretrained transformer models.

A1 represents conventional hard voting under the same outer ten-fold cross-validation protocol used for the remaining ablation variants. For each outer fold, the six classifiers were trained on the corresponding outer-training partition, their posterior probabilities for the untouched outer-validation partition were converted into binary decisions using the common threshold of 0.5, and the final class was determined by majority voting. The previous designation that A1 did not use out-of-fold probability estimation referred specifically to the absence of the inner probability-generation stage used for distance and weight estimation; it did not mean that A1 was evaluated without outer cross-validation.

Table 9. Ablation configurations used to evaluate the principal components of the proposed WDELM framework.

Variant	Inner OOF Probabilities Used in Ensemble Construction	Distance Measure	Weighting Strategy	Normalisation	Fusion Strategy
A1	✗	–	Equal contributions	✗	Majority voting over binary predictions
A2	✓	–	Equal contributions	✗	Majority voting over binary predictions
A3	✓	Euclidean	Distance-aware	✓	Weighted posterior-probability fusion
A4	✓	Cosine	Validation-accuracy-based	✓	Weighted posterior-probability fusion
A5	✓	Cosine	Distance-aware	✗	Unnormalised weighted posterior fusion
A6	✓	Cosine	Distance-aware	✓	Majority voting over binary predictions
A7	✓	–	Uniform, $w_i = 1/m$	✓	Equal-weight posterior-probability fusion
Full without XGBoost	✓	Cosine	Distance-aware	✓	Weighted posterior-probability fusion
Full without LGBM	✓	Cosine	Distance-aware	✓	Weighted posterior-probability fusion
Full without RF	✓	Cosine	Distance-aware	✓	Weighted posterior-probability fusion
Full without AdaBoost	✓	Cosine	Distance-aware	✓	Weighted posterior-probability fusion
Full without GB	✓	Cosine	Distance-aware	✓	Weighted posterior-probability fusion
Full without LR	✓	Cosine	Distance-aware	✓	Weighted posterior-probability fusion
Complete WDELM	✓	Cosine	Distance-aware	✓	Weighted posterior-probability fusion

Note: A7 denotes the equal-weight soft-voting baseline. It uses the same six classifiers, fold-specific posterior-probability predictions, outer cross-validation folds, posterior-probability fusion procedure, and decision threshold of 0.5 as the complete WDELM. Each classifier is assigned the uniform weight $w_i = 1/m = 1/6$; therefore, A7 differs from the complete WDELM only in the weight-determination strategy.

Table 10. Performance comparison of the proposed WDELM ablation variants and leave-one-classifier-out configurations under stratified ten-fold cross-validation. Precision, recall, and F1-score are reported for the Real class using `pos_label=1`.

Model	Real-Class Precision	Real-Class Recall	Real-Class F1-Score	Kappa	AUC	Accuracy
A1	88.881 ± 0.281	89.203 ± 0.217	89.042 ± 0.249	75.930 ± 0.558	95.053 ± 0.254	88.061 ± 0.276
A2	88.279 ± 1.502	90.217 ± 1.087	89.219 ± 0.236	76.043 ± 0.944	95.260 ± 0.198	88.140 ± 0.433
A3	88.348 ± 1.433	90.290 ± 1.159	89.289 ± 0.165	76.203 ± 0.784	95.293 ± 0.194	88.219 ± 0.355
A4	88.408 ± 1.373	90.290 ± 1.159	89.321 ± 0.134	76.285 ± 0.703	95.317 ± 0.183	88.258 ± 0.315
A5	92.748 ± 1.812	92.826 ± 2.111	92.756 ± 1.019	84.117 ± 2.213	97.125 ± 0.686	92.120 ± 1.097
A6	87.604 ± 1.542	90.725 ± 1.449	89.112 ± 0.099	75.614 ± 0.719	95.293 ± 0.193	87.943 ± 0.315
A7	92.114 ± 1.534	92.391 ± 1.846	92.247 ± 0.917	82.932 ± 1.967	96.884 ± 0.612	91.529 ± 0.944
Without XGBoost	89.768 ± 1.395	91.377 ± 0.739	90.560 ± 0.901	79.076 ± 2.110	96.299 ± 0.443	89.637 ± 1.036
Without LGBM	91.209 ± 1.255	91.449 ± 0.934	91.319 ± 0.535	80.936 ± 1.265	96.610 ± 0.347	90.544 ± 0.619
Without RF	90.000 ± 1.541	91.087 ± 1.137	90.532 ± 1.062	79.088 ± 2.412	96.088 ± 0.422	89.637 ± 1.190
Without AdaBoost	90.246 ± 1.963	91.377 ± 0.703	90.794 ± 0.931	79.640 ± 2.282	96.465 ± 0.369	89.913 ± 1.113
Without GB	91.247 ± 0.975	92.029 ± 0.725	91.632 ± 0.589	81.561 ± 1.351	96.933 ± 0.264	90.859 ± 0.665
Without LR	90.898 ± 0.721	91.812 ± 0.589	91.349 ± 0.280	80.922 ± 0.682	96.489 ± 0.421	90.544 ± 0.332
Complete WDELM	92.748 ± 1.812	92.826 ± 2.111	92.756 ± 1.019	84.117 ± 2.213	97.125 ± 0.686	92.120 ± 1.097

Note: Variant A5 omits only the final normalisation of the preliminary agreement scores. Because its output satisfies $\widetilde{\mathbf{P}}_{\mathrm{WDELM}} = (m-1)\mathbf{P}_{\mathrm{WDELM}}$, it preserves the ranking of the evaluated instances and therefore produces the same ROC-AUC as the complete normalised WDELM. The reported threshold-dependent metrics were obtained using the equivalent threshold $0.5(m-1) = 2.5$, which yields the same binary predictions as the normalised output with a threshold of 0.5. A7 represents equal-weight soft voting with $w_i = 1/6$ for all six classifiers.

A2 also applies majority voting over binary decisions, but it retains the inner out-of-fold posterior-probability generation stage used in the WDELM workflow before the classifier probabilities are converted into binary outputs. Thus, A1 and A2 differ in whether this inner probability-estimation stage is included, whereas both retain equal hard-voting fusion. Because A1 and A2 differ from the complete WDELM in several components simultaneously, neither comparison is used to isolate the contribution of distance-aware weighting.

To provide the one-factor comparison, A7 applies equal-weight soft voting to exactly the same six outer-validation posterior-probability vectors used by the complete WDELM. For outer fold r, the A7 ensemble posterior probability is defined as

$$\mathbf{P}^{(r)}_{\mathrm{A7,val}} = \frac{1}{m}\sum_{i=1}^{m}\mathbf{P}^{(r)}_{i,\mathrm{val}}, \qquad m = 6. \tag{27}$$

The same threshold of 0.5 is applied to both A7 and WDELM. Therefore, A7 differs from the complete WDELM only in the weight-determination rule: A7 uses uniform weights $w_i = 1/6$, whereas WDELM uses fold-specific distance-aware weights estimated exclusively from the corresponding outer-training partition. Variants A3 and A4 evaluate, respectively, Euclidean-distance weighting and fixed validation-accuracy-based weighting in place of the proposed cosine- distance strategy. Variant A6 replaces weighted posterior-probability fusion with majority voting while retaining the remaining WDELM components.

Variant A5 omits only the final normalisation of the preliminary agreement scores. Under the adopted formulation, its output satisfies

$$\widetilde{\mathbf{P}}_{\mathrm{WDELM}} = (m-1)\mathbf{P}_{\mathrm{WDELM}}. \tag{28}$$

Consequently, A5 is a scale-equivalent representation of the complete WDELM. The positive rescaling preserves the ranking of all evaluated instances and therefore does not change ROC-AUC. When the equivalent threshold $0.5(m-1) = 2.5$ is applied to the unnormalised output, its binary predictions and threshold-dependent metrics are also identical to those of the complete normalised WDELM. Thus, this ablation demonstrates that normalisation is required to preserve the probabilistic range and the conventional threshold of 0.5, but it is not an independent source of improved discrimination.

The comparison between A7 and the complete WDELM provides the most direct assessment of the proposed weighting mechanism. Both configurations use the same six classifiers, identical outer-fold assignments, the same fold-specific posterior-probability predictions, the same soft-fusion procedure, and the same decision threshold of 0.5. Their only difference is the method used to determine the classifier weights. Under the adopted stratified ten-fold cross-validation protocol, A7 achieved a mean accuracy of 91.529%, a mean ROC-AUC of 96.884%, and a mean Real-class F1-score of 92.247%, compared with 92.120%, 97.125%, and 92.756% for the complete WDELM, respectively. The corresponding improvements of 0.591, 0.241, and 0.509 percentage points provide a controlled estimate of the incremental empirical contribution of the proposed distance-aware weighting rule relative to equal-weight posterior-probability averaging within the evaluated dataset and validation protocol. Relative performance-reduction categories were defined according to the absolute decrease in accuracy relative to the complete WDELM configuration. A reduction of less than 2 percentage points was classified as Low, a reduction of 2 to less than 3 percentage points as Moderate, a reduction of 3 to less than 4 percentage points as High, and a reduction of 4 percentage points or more as Very High.

Table 11 further quantifies the performance change of each ablation configuration relative to the complete WDELM. A1, A2, A3, A4, and A6 alter several components and therefore describe the combined consequences of those design changes rather than the isolated contribution of the weighting rule. By contrast, A7 holds the classifiers, folds, posterior probabilities, fusion rule, and threshold constant and changes only the weights from fold-specific distance-aware values to $1/6$. Under the evaluated protocol, A7 is lower than WDELM by 0.591 percentage points in accuracy, 0.241 points in ROC-AUC, and 0.509 points in Real-class F1-score.

Variant A5 presents a special analytical case. Because the unnormalised output satisfies $\widetilde{\mathbf{P}}_{\mathrm{WDELM}} = (m-1)\mathbf{P}_{\mathrm{WDELM}}$, its performance differences relative to the complete WDELM are zero when the mathematically equivalent threshold $0.5(m-1) = 2.5$ is used. Therefore, the A5 row in Table 11 reports zero change for precision, recall, F1-score, Cohen's Kappa, ROC-AUC, and accuracy. This result confirms that normalisation preserves the probabilistic interpretation and the standard decision threshold, but does not independently improve discriminative performance.

Table 11. Performance differences relative to the complete WDELM configuration under the evaluated ablation settings. Differences in precision, recall, and F1-score refer to the Real class.

Ablation Variant	ΔReal Precision	ΔReal Recall	ΔReal F1-Score	ΔKappa	ΔAUC	ΔAccuracy	Relative Performance Reduction
A1	−3.867	−3.623	−3.714	−8.187	−2.072	−4.059	Very High
A2	−4.469	−2.609	−3.537	−8.074	−1.865	−3.980	High
A3	−4.400	−2.536	−3.467	−7.914	−1.832	−3.901	High
A4	−4.340	−2.536	−3.435	−7.832	−1.808	−3.862	High
A5	0.000	0.000	0.000	0.000	0.000	0.000	None †
A6	−5.144	−2.101	−3.644	−8.503	−1.832	−4.177	Very High
A7	−0.634	−0.435	−0.509	−1.185	−0.241	−0.591	Low ‡
Without XGBoost	−2.980	−1.449	−2.196	−5.041	−0.826	−2.483	Moderate
Without LGBM	−1.539	−1.377	−1.437	−3.181	−0.515	−1.576	Low
Without RF	−2.748	−1.739	−2.224	−5.029	−1.037	−2.483	Moderate
Without AdaBoost	−2.502	−1.449	−1.962	−4.477	−0.660	−2.207	Moderate
Without GB	−1.501	−0.797	−1.124	−2.556	−0.192	−1.261	Low
Without LR	−1.850	−1.014	−1.407	−3.195	−0.636	−1.576	Low

† Variant A5 is mathematically equivalent to the complete normalised WDELM up to multiplication by the positive constant $m - 1$. Its threshold-dependent results are therefore identical when the equivalent decision threshold $0.5(m - 1)$ is used. ‡ A7 represents equal-weight soft voting with $w_i = 1/6$ for all six classifiers.

The leave-one-classifier-out results in Table 11 show that removing XGBoost or RF produced the largest accuracy reductions among the individual classifier ablations. In both cases, accuracy decreased by 2.483 percentage points. Removing AdaBoost caused an accuracy reduction of 2.207 percentage points. Removing LGBM or LR reduced accuracy by 1.576 percentage points, whereas removing GB produced the smallest accuracy reduction of 1.261 percentage points.

The F1-score results show a similar pattern. Removing RF caused the largest F1-score reduction among the leave-one-classifier-out configurations (2.224 percentage points), followed by removing XGBoost (2.196 percentage points) and AdaBoost (1.962 percentage points). By contrast, removing GB resulted in the smallest F1-score reduction (1.124 percentage points). The ROC-AUC reductions were also largest after removing RF (1.037 percentage points) and XGBoost (0.826 percentage points), while removing GB produced the smallest ROC-AUC decrease (0.192 percentage points).

Accordingly, Tables 10 and 11 consistently indicate that XGBoost and RF made the largest empirical contributions to maintaining the performance of the complete ensemble, whereas removing GB produced the smallest overall change.

The reported ablation results describe empirical performance changes under the adopted internal cross-validation protocol. Because the archived aggregate outputs did not retain the fold-specific cumulative distance values and intermediate fitted weights, no retrospective numerical interpretation is made regarding the exact deviation of each classifier weight from $1/6$.

4.4. Error Analysis

The pooled outer-validation results reported in this section were obtained by concatenating the final predictions generated for the ten independently evaluated outer-validation folds. For every outer fold, the WDELM weights had already been estimated from the corresponding outer-training partition and were frozen before the outer-validation predictions were generated. Consequently, the pooled prediction vector was used only for aggregate reporting and was not used to calculate a global weight vector.

The confusion matrix in Figure 7 summarises the classification results obtained for the complete set of 2538 evaluated instances. The proposed WDELM correctly classified

1057 fake-news instances and 1281 real-news instances. It produced 101 false-negative predictions, in which fake-news instances were classified as real, and 99 false-positive predictions, in which real-news instances were classified as fake.

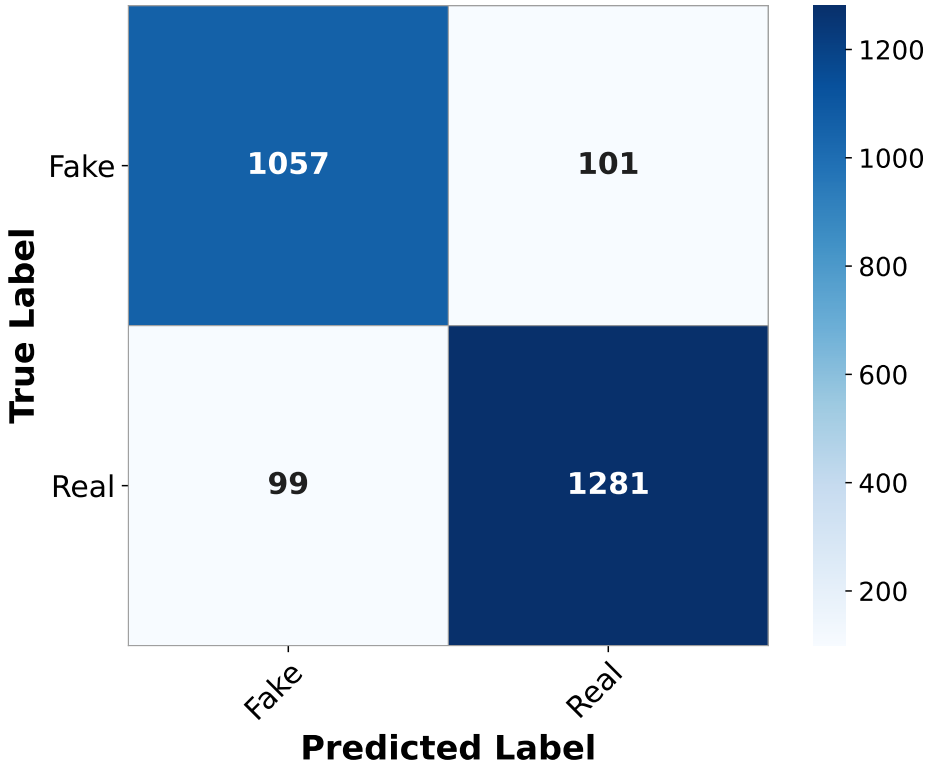

Figure 7. Confusion matrix of the proposed WDELM constructed from the concatenated predictions generated across the ten outer-validation folds. For each fold, the WDELM weights were estimated exclusively from the corresponding outer-training partition and frozen before evaluation of the associated outer-validation partition.

The model generated 2338 correct predictions and 200 incorrect predictions, corresponding to an overall accuracy of approximately 92.12%. When the Fake class is treated as the target class, the concatenated outer-validation predictions yield a precision of 91.436%, a recall of 91.278%, and an F1-score of 91.357%. When the Real class is treated as the target class, the corresponding values are 92.692%, 92.826%, and 92.759%, respectively. Equal weighting of the two classes produces a macro precision of 92.064%, a macro recall of 92.052%, and a pooled macro-F1 score of 92.058%, as reported in Table 12.

Table 12. Class-specific and macro-averaged performance calculated from the concatenated predictions of the ten separately evaluated outer-validation folds. The fold-specific WDELM weights were estimated exclusively within the corresponding outer-training partitions.

Evaluation Scope	Precision (%)	Recall (%)	F1-Score (%)
Fake class	91.436	91.278	91.357
Real class	92.692	92.826	92.759
Macro average	92.064	92.052	92.058

Note: The values were calculated from the confusion matrix formed by concatenating the predictions from the ten separately evaluated outer-validation folds, containing 2538 predictions. They are therefore pooled estimates and are distinct from the fold-level mean ± standard deviation values reported in the cross-validation tables.

The close class-specific F1-scores and the similar numbers of false-positive and false-negative predictions indicate relatively balanced performance between fake and real news under the adopted evaluation setting. Importantly, the class-wise values in Table 12 were calculated from the complete concatenated set of outer-validation predictions, whereas the values reported as mean ± standard deviation in the cross-validation tables were calculated separately for each fold.

The confusion matrix was constructed by concatenating the predictions generated across the ten outer-validation folds. Although each outer-validation partition was evaluated separately using fold-specific models and frozen WDELM weights, the resulting fold-level estimates should not be regarded as statistically independent because the corresponding outer-training partitions overlap. In each fold, the vocabulary, feature transformation, base classifiers, pairwise distances, and WDELM weights were determined without access to the corresponding outer-validation partition. Therefore, the matrix summarises strictly fold-wise cross-validated classification behaviour across the evaluated dataset rather than the performance of a single representative fold. Nevertheless, these results remain specific to the studied dataset and do not independently establish generalisability to unseen domains, dialects, time periods, or real-world deployment conditions.

The ROC curve in Figure 8 illustrates the discriminative performance of the proposed WDELM framework based on the concatenated probability predictions generated for the ten untouched outer-validation folds. The resulting ROC-AUC is 0.97125, indicating that the proposed model effectively distinguishes between fake and real Arabic news within the evaluated dataset and experimental protocol. The curve remains above the random-classifier reference line across the evaluated operating range, indicating favorable discrimination between the two classes.

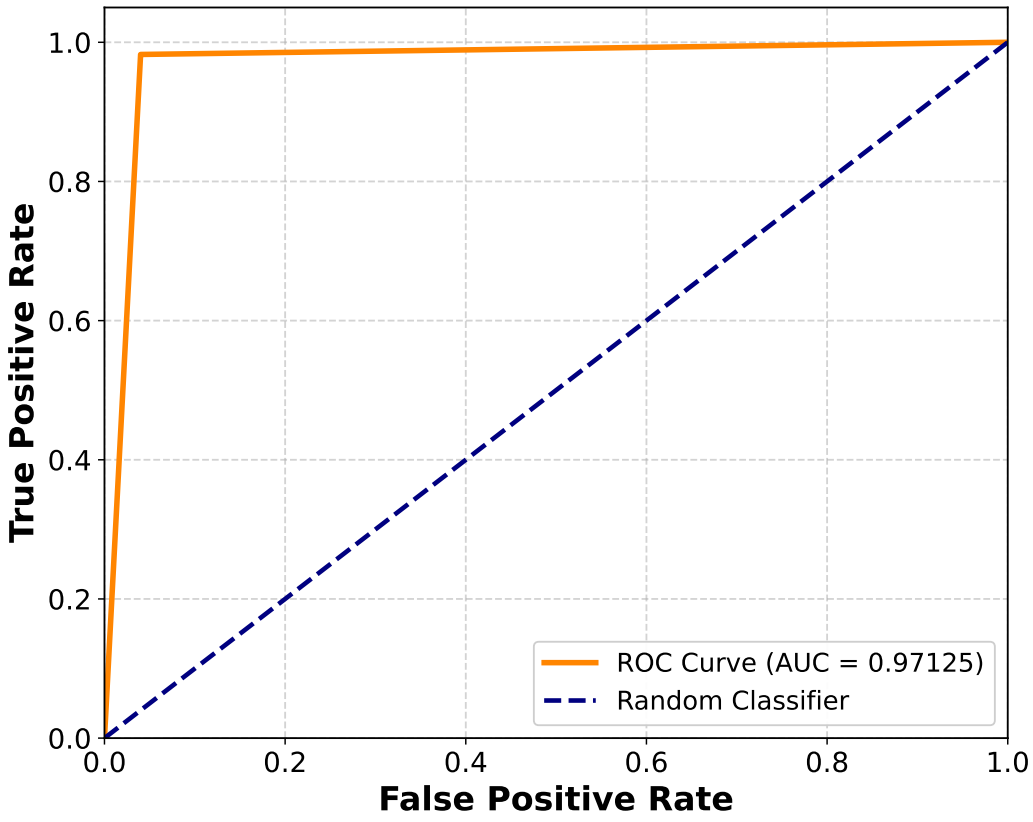

Figure 8. ROC curve of the proposed WDELM framework based on the concatenated outer-validation probabilities generated under the strictly fold-wise stratified ten-fold cross-validation protocol.

Table 13 presents representative misclassified Arabic news samples and illustrates several recurring sources of classification error. Some fake-news samples were classified as real when their texts contained factual terminology, named entities, or weak lexical indicators of misinformation. Conversely, several real-news samples were classified as fake when they contained sensational wording, controversial expressions, or structures resembling rumor-oriented discourse. Although ISRI stemming reduces morphological variation, it may also remove distinctions that contribute to contextual interpretation. These examples indicate that reliance on lexical representations can limit discrimination when fake and real content share similar vocabulary.

Table 13. Representative misclassified samples and linguistic challenges affecting Arabic fake news detection performance.

ISRI Root Stemmed Text	Original/Normalised Text	Actual Class	Predicted Class
حدث صين خفافيش يلم كونتيج تنب ب كورو قبل سنو	تحدث عن الصين والخفافيش.. فيلم « كونتيجن » تنبأ بـ «# كورونا » قبل 9 سنوات	Fake	Real
يقل صين قتل صاب كورو	يقولون الصينيين يقتلون المصابين بكورونا	Fake	Real
هجم بعض سرب طعر غرب شرع صين	هجوم البعوض وأسراب طائر الغر اب في شوارع الصين	Fake	Real
خبر نشر خصص صبه تشل كسس شيز ب يرس كورو غير صحح	الأخبار المنتشرة بخصوص إصابة التشيلي أليكسيس سانشيز بفيروس كورونا غير صحيحة	Real	Fake
علم تشل يثر جدل حول صبه كسس شيز يرس كورو	الإعلام التشيلي يثير الجدل حول إصابة أليكسيس سانشيز بفيروس # كورونا	Real	Fake
هل يتم هاء عام درس منح افد سبب كورو	هل يتم انهاء العام الدراسي ومنح إفادات بسبب كورونا ؟ «# كورونا »	Real	Fake

Note: For clarity, English translations of the Arabic original/normalised texts are provided below in the same order as the misclassified samples reported in the table. **Sample 1:** "The film *Contagion* discussed China and bats and predicted coronavirus nine years earlier." **Sample 2:** "They say that Chinese authorities are killing people infected with coronavirus." **Sample 3:** "An attack involving mosquitoes and flocks of birds in the streets of China." **Sample 4:** "The circulating reports concerning the infection of Alexis Sánchez with coronavirus are incorrect." **Sample 5:** "Chilean media reports have generated controversy regarding the alleged infection of Alexis Sánchez with coronavirus." **Sample 6:** "Will the academic year be ended and certificates granted because of coronavirus?" The Arabic entries in the "ISRI Root Stemmed Text" column represent algorithmically reduced root/stem forms generated during preprocessing and are therefore not grammatically complete Arabic sentences. Accordingly, the English translations above correspond to the semantic content of the associated original/normalised Arabic texts rather than to a literal token-by-token translation of the stemmed representations.

The incorrectly classified samples help identify the proposed model's limitations. Some fake news samples were predicted as real because they contained factual terms and COVID-19-related entities, making them semantically similar to authentic news. Conversely, several real news samples were classified as fake due to sensational wording, controversial expressions, and rumor-like sentence structures.

The occurrence of COVID-19-related vocabulary does not conflict with the reported collection period. January 2024–October 2025 refers to the publication or retrieval dates of the collected posts, whereas their textual content may refer to earlier pandemic events, vaccination claims, retrospective public-health reports, or misinformation that continued to circulate after the initial pandemic period. The publication date of a post and the historical date of the event discussed in its content are therefore distinct variables.

The misclassification results indicate that the main challenges include ambiguous wording, short text length, named-entity similarity, and overlap between fake- and real-news vocabulary. In addition, interrogative sentences, dialectal Arabic, and light stemming may reduce contextual understanding and increase classification difficulty. These findings suggest that incorporating deeper contextual and semantic representations could improve the robustness of Arabic fake news detection models.

4.5. Comparative Analysis with Baseline Models

Table 14 presents the performance comparison between the proposed WDELM and eleven baseline machine learning models using stratified cross-validation. The results are

reported as the mean ± standard deviation across all folds to evaluate both predictive performance and model stability. As shown in the table, WDELM achieved the best overall results, with a precision of 92.748 ± 1.812%, recall of 92.826 ± 2.111%, a fold-averaged Real-class F1-score of 92.756 ± 1.019%, Cohen's Kappa of 84.117 ± 2.213%, ROC-AUC of 97.125 ± 0.686%, and accuracy of 92.120 ± 1.097%. These results indicate that WDELM provides the most balanced overall classification performance among the evaluated models under the adopted stratified cross-validation protocol.

Table 14. Comparative performance of WDELM and the baseline machine learning models under stratified ten-fold cross-validation, reported as mean ± standard deviation. Precision, recall, and F1-score correspond to the Real class using `pos_label=1`.

Model	Real-Class Precision	Real-Class Recall	Real-Class F1-Score	Kappa	ROC-AUC	Accuracy
LR	89.710 ± 1.637	92.609 ± 1.478	91.125 ± 1.207	80.166 ± 2.740	96.220 ± 0.592	90.190 ± 1.351
XGBoost	89.744 ± 0.978	89.928 ± 0.983	89.830 ± 0.669	77.683 ± 1.453	95.290 ± 0.628	88.929 ± 0.721
RF	89.967 ± 0.757	91.522 ± 1.912	90.724 ± 0.947	79.482 ± 1.864	96.395 ± 0.368	89.835 ± 0.941
GB	85.105 ± 3.454	90.362 ± 2.506	87.561 ± 1.036	71.619 ± 3.126	93.684 ± 0.826	86.013 ± 1.472
LGBM	87.877 ± 1.227	88.623 ± 1.646	88.236 ± 1.024	74.093 ± 2.188	94.403 ± 0.773	87.156 ± 1.088
AdaBoost	78.796 ± 8.829	71.449 ± 18.457	72.421 ± 5.601	43.578 ± 1.602	82.481 ± 1.636	71.867 ± 1.145
ANN	91.349 ± 1.927	92.681 ± 0.983	91.989 ± 0.667	82.263 ± 1.726	96.583 ± 0.564	91.214 ± 0.834
SGD	54.717 ± 8.663	85.217 ± 10.195	66.543 ± 9.148	−0.327 ± 28.397	45.274 ± 27.150	53.029 ± 13.180
Bagging	88.743 ± 1.162	87.246 ± 1.881	87.973 ± 1.089	73.919 ± 2.166	93.958 ± 0.995	87.038 ± 1.089
RC	52.831 ± 6.631	77.899 ± 10.911	62.942 ± 8.193	−5.058 ± 23.104	45.646 ± 24.347	50.230 ± 10.939
ET	91.072 ± 1.217	92.101 ± 1.221	91.571 ± 0.559	81.397 ± 1.275	97.016 ± 0.173	90.780 ± 0.627
WDELM	92.748 ± 1.812	92.826 ± 2.111	92.756 ± 1.019	84.117 ± 2.213	97.125 ± 0.686	92.120 ± 1.097

Among the baseline classifiers, ANN achieved the strongest overall performance, with an F1-score of 91.989% and Accuracy of 91.214%, followed by ET, which obtained an F1-score of 91.571%, Accuracy of 90.780%, and a ROC-AUC of 97.016%. LR and RF also produced competitive results, whereas GB, AdaBoost, SGD, and RC showed noticeably lower performance. Compared with ANN, which achieved the strongest overall performance among the baseline models, WDELM increased the F1-score by 0.767 percentage points, accuracy by 0.906 percentage points, and ROC-AUC by 0.542 percentage points under the adopted cross-validation protocol.

The observed performance of WDELM is consistent with its distance-aware adaptive weighting mechanism, which assigns larger weights to classifiers whose posterior probability patterns exhibit stronger agreement with the remaining ensemble members and smaller weights to classifiers with more divergent prediction patterns. Unlike equal-weight voting methods, WDELM explicitly incorporates inter-classifier relationships into the probabilistic fusion process. In addition, the relatively small standard deviations across the evaluation metrics indicate stable performance across the examined cross-validation folds. Nevertheless, broader generalisation beyond the evaluated dataset requires validation under additional domains, dialects, time periods, and external datasets.

Table 15 provides a descriptive summary of fold-to-fold variability for the principal performance metrics of the complete WDELM under stratified ten-fold cross-validation. The intervals were calculated from the ten observed outer-fold estimates using the Student's t-critical value with nine degrees of freedom. The descriptive 95% fold-based intervals were $[91.335, 92.905]$% for accuracy, $[92.027, 93.485]$% for the Real-class F1-score, and $[96.634, 97.616]$% for ROC-AUC. These ranges summarise the observed variation among the ten folds. Because the folds share overlapping training partitions, they are correlated and do not represent ten independent observations. The intervals are therefore not interpreted as conventional population-level confidence intervals or as corrected estimates of external generalisation uncertainty.

Table 16 summarises the descriptive ranking of the proposed WDELM and the baseline machine learning models by averaging their positions across the six reported evaluation metrics. WDELM ranked first for all six metrics and obtained the best average rank of 1.00. Among the baseline methods, ANN achieved the strongest overall position with an average rank of 2.17, followed by ET with an average rank of 3.00. LR and RF obtained average ranks of 4.33 and 4.67, respectively, whereas XGBoost, LGBM, bagging, and GB occupied intermediate positions. AdaBoost, SGD, and ridge classifier (RC) obtained the lowest aggregate rankings. These descriptive ranks summarize the relative performance of the evaluated methods under the adopted cross-validation protocol and should not be interpreted independently as evidence of statistical significance.

Table 15. Mean performance and descriptive 95% fold-based intervals of the complete WDELM under stratified ten-fold cross-validation.

Metric	Mean (%)	Standard Deviation	Descriptive 95% Fold-Based Interval (%)
Real-class Precision	92.748	1.812	[91.452, 94.044]
Real-class Recall	92.826	2.111	[91.316, 94.336]
Real-class F1-score	92.756	1.019	[92.027, 93.485]
Cohen's Kappa	84.117	2.213	[82.534, 85.700]
ROC-AUC	97.125	0.686	[96.634, 97.616]
Accuracy	92.120	1.097	[91.335, 92.905]

Note: The intervals were calculated as $\bar{x} \pm t_{0.975,9} s / \sqrt{10}$ from the ten observed outer-fold estimates. Because cross-validation folds share overlapping training data, the fold-level values are correlated and do not constitute ten independent observations. These quantities are therefore reported only as descriptive 95% fold-based intervals and should not be interpreted as conventional population-level confidence intervals or as corrected estimates of external generalisation uncertainty. Precision, recall, and F1-score refer to the Real class using `pos_label=1`.

Table 16. Overall descriptive ranking of the proposed WDELM and baseline machine learning models based on six evaluation metrics.

Model	Real-Class Precision Rank	Real-Class Recall Rank	Real-Class F1-Score Rank	Kappa Rank	ROC-AUC Rank	Accuracy Rank	Average Rank
WDELM	1	1	1	1	1	1	1.00
ANN	2	2	2	2	3	2	2.17
ET	3	4	3	3	2	3	3.00
LR	6	3	4	4	5	4	4.33
RF	4	5	5	5	4	5	4.67
XGBoost	5	7	6	6	6	6	6.00
LGBM	8	8	7	7	7	7	7.33
Bagging	7	9	8	8	8	8	8.00
GB	9	6	9	9	9	9	8.50
AdaBoost	10	12	10	10	10	10	10.33
SGD	11	10	11	11	12	11	11.00
RC	12	11	12	12	11	12	11.67

4.6. Deep Learning-Based Comparative Analysis

To further validate the effectiveness of the proposed WDELM, a comparative analysis with several DL models was conducted. The dataset was first partitioned into 70% training data and 30% testing data using stratified sampling to preserve the class distribution. The results reported in this section were obtained using this independent stratified 70–30% train–test split to ensure that all deep learning architectures, pretrained transformer models, and the proposed WDELM were evaluated under the same experimental protocol. Therefore, these values differ slightly from the mean results obtained through stratified ten-fold cross-validation in the preceding sections and should not be interpreted

as outcomes from the same evaluation setting. All DL models and the proposed WDELM were trained exclusively on the training set, while the independent testing set was reserved for the final performance evaluation. DL approaches have demonstrated strong capabilities in text classification tasks by capturing semantic and contextual representations. In this study, representative deep learning architectures, including a convolutional neural network (CNN), long short-term memory (LSTM), bidirectional LSTM (BiLSTM), gated recurrent unit (GRU), CNN–LSTM, VGG, and ResNet, were implemented and evaluated using the same dataset and performance metrics. Table 17 presents the performance comparison between DL models and the proposed WDELM approach across multiple evaluation metrics.

Table 17. Performance comparison of deep learning architectures, pretrained Arabic transformer models, and the proposed WDELM under the common stratified 70–30% train–test protocol. All values are reported as percentages.

Algorithm	Real-Class Precision	Real-Class Recall	Real-Class F1-Score	Kappa	ROC-AUC	Accuracy
CNN	89.683	93.582	91.591	81.108	96.393	90.664
LSTM	83.549	93.582	88.281	72.486	94.633	86.502
BiLSTM	90.638	88.199	89.402	77.164	95.520	88.639
GRU	89.006	87.164	88.075	74.210	94.693	87.177
CNN–LSTM	91.342	87.371	89.312	77.200	94.714	88.639
VGG	83.734	93.789	88.477	72.945	94.536	86.727
ResNet	88.605	93.375	90.927	79.495	96.084	89.876
MARBERT	88.300	90.300	89.300	76.100	94.300	88.200
CAMeLBERT	89.900	87.000	88.400	75.100	94.600	87.600
QARiB	87.400	87.400	87.400	72.300	94.000	86.300
ArabicBERT	84.800	91.100	87.800	72.100	93.900	86.300
AraELECTRA	86.600	88.400	87.500	72.300	93.100	86.300
MARBERTv2	86.600	88.000	87.300	71.800	93.600	86.100
AraBERTv2	82.800	85.900	84.300	65.000	91.200	82.700
WDELM	91.962	94.647	93.285	85.173	96.933	92.651

To ensure a fair comparison, all DL models (CNN, LSTM, BiLSTM, GRU, CNN–LSTM, VGG, and ResNet) were trained using the same experimental settings. Each model employed a trainable embedding layer with an embedding dimension of 300 and a maximum sequence length of 300 tokens using a common vocabulary. The output layer used a sigmoid activation function with binary cross-entropy loss and the Adam optimiser (learning rate = 0.001). All models were trained for 15 epochs with a batch size of 32 and a dropout rate of 0.5. A dense layer of 128 neurons was used for CNN, LSTM, BiLSTM, GRU, and CNN–LSTM, while VGG and ResNet employed 256 neurons.

All transformer-based models, including MARBERT, MARBERTv2, CAMeLBERT, AraBERTv2, ArabicBERT, AraELECTRA, and QARiB, were fine-tuned under the same experimental settings to ensure a fair comparison. Each model used its official tokenizer and publicly available pretrained weights, with a maximum sequence length of 256 tokens represented by token IDs and attention masks. End-to-end fine-tuning was performed using a fully connected classification layer ($768 \rightarrow 2$) with two output units and a softmax activation function. The models were optimised using AdamW with a learning rate of 2×10^{-5}, a weight decay of 0.01, and a warm-up ratio of 0.1. Training was conducted for 5 epochs with a batch size of 32, a dropout rate of 0.1, and a fixed random seed of 42 to ensure reproducibility.

Under the common stratified 70–30% train–test protocol, the proposed WDELM achieved higher values than the evaluated deep learning and pretrained transformer models across all six reported metrics. The observed performance is consistent with the intended role of the distance-aware adaptive weighting mechanism, which incorporates

inter-model agreement into the probability-fusion process. However, the comparative results alone do not establish that this mechanism is the sole cause of the observed differences. The common stratified 70–30% train–test split and consistent evaluation procedure improved the comparability of the reported results across the examined methods.

To provide a comprehensive comparison with contemporary Arabic text classification methods, seven state-of-the-art pretrained Arabic transformer models were also fine-tuned and evaluated under the same stratified 70–30% train–test protocol. As shown in Table 17, MARBERT achieved the highest F1-score (89.300%), while CAMeLBERT obtained the highest ROC-AUC (94.600%). Nevertheless, the proposed WDELM achieved the highest reported performance across the evaluated metrics under the adopted experimental protocol, achieving the highest precision (91.962%), recall (94.647%), F1-score (93.285%), Kappa (85.173%), ROC-AUC (96.933%), and accuracy (92.651%). Compared with CNN, which achieved the strongest aggregate performance among the conventional deep learning models, WDELM increased the F1-score by 1.694 percentage points (93.285% vs. 91.591%), accuracy by 1.987 percentage points (92.651% vs. 90.664%), and ROC-AUC by 0.540 percentage points (96.933% vs. 96.393%). Compared with MARBERT, which achieved the strongest aggregate performance among the pretrained transformer models, WDELM increased the F1-score by 3.985 percentage points (93.285% vs. 89.300%), accuracy by 4.451 percentage points (92.651% vs. 88.200%), and ROC-AUC by 2.633 percentage points (96.933% vs. 94.300%). Under the adopted stratified 70–30% train–test protocol, WDELM obtained higher reported metric values than the evaluated standalone deep learning and pretrained transformer models. These results indicate competitive empirical performance within the examined dataset but do not establish universal superiority or external generalisability.

Table 18 summarises the overall ranking of the proposed WDELM, deep learning architectures, and pretrained Arabic transformer models by averaging their ranks across the six reported evaluation metrics. Tied performance values were assigned their corresponding average ranks. WDELM achieved the best overall position, ranking first for all six metrics and obtaining an average rank of 1.00. CNN ranked second overall with an average rank of 2.75, followed by ResNet with an average rank of 4.00. BiLSTM and CNN–LSTM obtained average ranks of 4.92 and 5.42, respectively. Among the pretrained transformer models, MARBERT achieved the strongest overall position with an average rank of 7.17, followed by CAMeLBERT with an average rank of 8.00. AraBERTv2 obtained the lowest overall position, with an average rank of 15.00. These descriptive rankings indicate that WDELM achieved the strongest aggregate performance under the adopted stratified 70–30% train–test protocol; however, they should not be interpreted as a statistical significance analysis.

4.7. Statistical Validation and Comprehensive Significance Analysis

To assess whether the observed performance differences among the evaluated machine learning methods were systematic across the ten matched cross-validation folds, a non-parametric statistical analysis was conducted. The Friedman test was first applied to test the global null hypothesis that all methods had equivalent performance distributions across the matched folds. Kendall's coefficient of concordance, W, was additionally calculated as an effect-size measure for the Friedman test to quantify the magnitude of the overall differences among methods.

When the Friedman test indicated a statistically significant global difference, pairwise comparisons between WDELM and the baseline methods were performed using the Wilcoxon signed-rank test. Holm's sequential correction was applied to control the family-wise error rate across multiple comparisons. A Nemenyi critical-difference di-

agram was also used to visualize groups of methods whose average ranks were not significantly different.

Table 18. Overall performance ranking of WDELM, deep learning architectures, and pretrained Arabic transformer models under the common stratified 70–30% train–test protocol.

Algorithm	Real-Class Precision Rank	Real-Class Recall Rank	Real-Class F1-Score Rank	Kappa Rank	ROC-AUC Rank	Accuracy Rank	Average Rank
WDELM	1.0	1.0	1.0	1.0	1.0	1.0	1.00
CNN	5.0	3.5	2.0	2.0	2.0	2.0	2.75
ResNet	7.0	5.0	3.0	3.0	3.0	3.0	4.00
BiLSTM	3.0	9.0	4.0	5.0	4.0	4.5	4.92
CNN–LSTM	2.0	12.0	5.0	4.0	5.0	4.5	5.42
MARBERT	8.0	7.0	6.0	6.0	10.0	6.0	7.17
CAMeLBERT	4.0	14.0	8.0	7.0	8.0	7.0	8.00
VGG	13.0	2.0	7.0	9.0	9.0	9.0	8.17
GRU	6.0	13.0	10.0	8.0	6.0	8.0	8.50
LSTM	14.0	3.5	9.0	10.0	7.0	10.0	8.92
ArabicBERT	12.0	6.0	11.0	13.0	12.0	12.0	11.00
QARiB	9.0	11.0	13.0	11.5	11.0	12.0	11.25
AraELECTRA	10.5	8.0	12.0	11.5	14.0	12.0	11.33
MARBERTv2	10.5	10.0	14.0	14.0	13.0	14.0	12.58
AraBERTv2	15.0	15.0	15.0	15.0	15.0	15.0	15.00

Note: For each evaluation metric, methods with identical performance values were assigned the average of the ranks that they would otherwise occupy. The overall average rank was then calculated as the arithmetic mean of the six metric-specific ranks.

The ten folds were treated as matched evaluation blocks because all methods were evaluated using the same fold assignments. However, cross-validation folds are correlated performance estimates derived from one dataset and are not equivalent to ten independent datasets. Moreover, a Wilcoxon signed-rank test based on only ten paired observations has limited statistical power and limited attainable p-value resolution. Therefore, the pairwise inferential results are interpreted as complementary evidence within the adopted cross-validation protocol rather than definitive evidence of superiority or external generalisability.

Table 19 presents the Friedman test results and Kendall's W effect sizes across the 12 evaluated machine learning methods. Statistically significant global differences were observed for all six metrics. However, the effect-size estimates show that the magnitude of these differences varied across metrics.

Table 19. Friedman test results and Kendall's W effect sizes across the evaluated machine learning methods under stratified ten-fold cross-validation.

Metric	Methods	Blocks	Degrees of Freedom	χ^2	p-Value	Kendall's W	Magnitude
Real-class Precision	12	10	11	79.602	1.76×10^{-12}	0.724	Strong
Real-class Recall	12	10	11	42.380	1.39×10^{-5}	0.385	Moderate
Real-class F1-score	12	10	11	88.031	4.05×10^{-14}	0.800	Strong
Cohen's Kappa	12	10	11	88.038	4.04×10^{-14}	0.800	Strong
ROC-AUC	12	10	11	96.354	9.39×10^{-16}	0.876	Strong
Accuracy	12	10	11	89.074	2.53×10^{-14}	0.810	Strong

Note: Kendall's coefficient of concordance was calculated as $W = \chi^2/[N(k-1)]$, where $N = 10$ matched folds and $k = 12$ methods. Values closer to zero indicate weaker agreement in method rankings, whereas values closer to one indicate stronger and more consistent differences among the evaluated methods. The magnitude labels are descriptive and should be interpreted in conjunction with the metric-specific results.

The largest Kendall's W value was observed for ROC-AUC ($W = 0.876$), followed by accuracy ($W = 0.810$), Real-class F1-score ($W = 0.800$), and Cohen's Kappa ($W = 0.800$). Real-class precision also showed a strong overall effect ($W = 0.724$). Recall produced the smallest effect-size estimate ($W = 0.385$), indicating that the methods were less consistently separated with respect to recall than for the other metrics.

These results indicate systematic differences in the relative performance of the evaluated methods across the matched folds. Nevertheless, the significant global Friedman tests do not establish that WDELM significantly outperformed every individual baseline; pairwise comparisons are required for that interpretation.

Following the significant Friedman tests, pairwise Wilcoxon signed-rank comparisons with Holm correction were used to compare WDELM with each baseline method. The adjusted p-values are reported in Table 20. Statistically significant differences were observed between WDELM and several lower-performing baseline methods. Nevertheless, no statistically significant difference was detected between WDELM and ET, ANN, RF, or LR for most metrics after multiple-comparison correction.

Table 20. Holm-adjusted p-values from pairwise Wilcoxon signed-rank comparisons between WDELM and the baseline machine learning methods across ten matched cross-validation folds.

Compared Model	Accuracy (Adj. p)	Precision (Adj. p)	Recall (Adj. p)	F1-Score (Adj. p)	Kappa (Adj. p)
LR	0.109	0.096	1.000	0.109	0.148
XGBoost	0.049	0.137	0.297	0.068	0.049
RF	0.182	0.480	1.000	0.146	0.193
GB	0.041	0.117	0.098	0.041	0.041
LGBM	0.041	0.078	0.342	0.041	0.041
AdaBoost	0.021	0.021	0.342	0.021	0.021
ANN	0.387	0.984	1.000	0.387	0.387
SGD	0.021	0.021	1.000	0.021	0.021
Bagging	0.031	0.148	0.123	0.031	0.031
RC	0.021	0.021	0.043	0.021	0.021
ET	0.387	0.984	1.000	0.387	0.387

Accordingly, the pairwise results do not support a claim that WDELM was statistically superior to every leading baseline. Instead, they indicate that WDELM obtained the highest mean values while remaining statistically comparable with several strong competing methods under the adopted cross-validation protocol.

The Wilcoxon results should also be interpreted cautiously because only ten matched fold-level observations were available. This small number limits both the power of the test and the resolution of attainable p-values. Furthermore, the folds were derived from a single dataset and are not fully independent observations.

Kendall's W was reported as the omnibus effect-size measure accompanying the Friedman tests. For the pairwise Wilcoxon signed-rank comparisons, the signed fold-level differences and signed-rank statistics were retained and used to calculate the corresponding pairwise effect sizes. These effect sizes, together with the Wilcoxon statistics, unadjusted p-values, Holm-adjusted p-values, and the direction of each paired comparison, are reported. Reporting both multiplicity-adjusted significance levels and effect-size estimates provides a more complete description of the magnitude and consistency of the observed paired performance differences under the adopted cross-validation protocol.

Figure 9 illustrates the critical difference (CD) diagram obtained from the Nemenyi post-hoc analysis. The calculated CD value was 5.190, indicating the minimum difference in average ranks required for two classifiers to be considered significantly different at the 5% significance level. The proposed WDELM achieved the best overall average rank

(2.4) among all evaluated machine learning models, followed closely by ET (2.7) and the ANN (3.0). These classifiers formed the leading statistical group, indicating that although WDELM obtained the highest ranking, its performance was statistically comparable with the strongest competing ensemble methods.

In contrast, AdaBoost, SGD, and RC obtained substantially higher average ranks, reflecting consistently inferior predictive performance across the cross-validation folds. Their separation from the leading models in the CD diagram indicates statistically meaningful performance differences, corroborating the results of the Wilcoxon signed-rank tests. The CD analysis places WDELM within the leading statistical group together with ET and ANN. Although WDELM obtained the best average rank, the diagram does not indicate a statistically significant separation from all leading competitors.

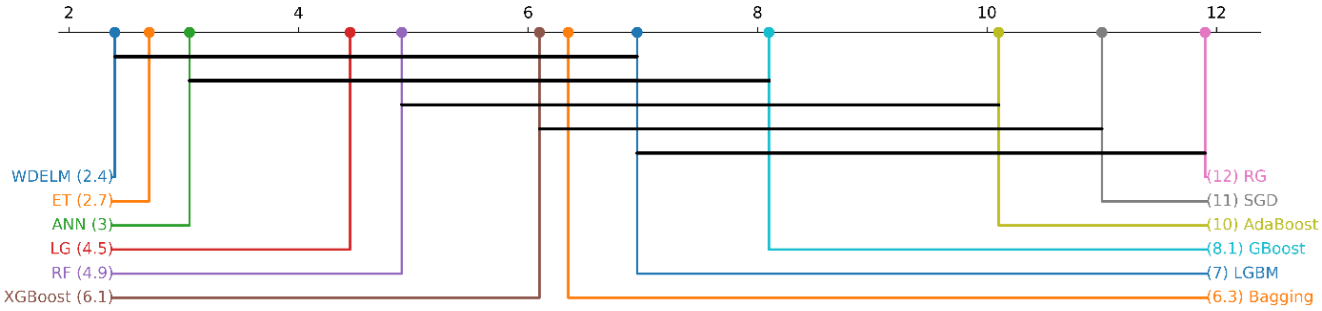

Figure 9. CD diagram based on the Nemenyi post-hoc test.

4.8. Explainable AI (XAI) Analysis

To enhance the transparency and interpretability of the proposed WDELM, XAI analysis was conducted using the Local Interpretable Model-Agnostic Explanations (LIME) and SHapley Additive exPlanations (SHAP). Explainability has become a critical requirement in fake news detection systems, particularly in high-stakes domains involving political, social, and public-health misinformation, where understanding model decisions is as important as predictive accuracy.

The explainability analysis was performed on the final predictions produced by the proposed WDELM ensemble. Since WDELM generates posterior probabilities by adaptively combining the normalised probability outputs of the constituent classifiers, both SHAP and LIME explain the final ensemble prediction rather than the predictions of individual base learners. SHAP explanations were generated using the KernelExplainer with a representative subset of the training data as the background reference distribution. Global feature importance was computed using the mean absolute SHAP value across all testing instances. LIME explanations were generated using LimeTextExplainer. For each instance, perturbed samples were generated around the original document and the locally weighted surrogate model was fitted to approximate the WDELM prediction. The reported LIME explanations were selected to illustrate representative correctly classified and misclassified test instances with different local contribution profiles.

The class-wise feature analysis identified lexical features with comparatively larger model-derived importance values for the Fake and Real outputs. These quantities describe associations within the fitted model and evaluated corpus; they do not establish that the identified tokens possess class-specific meanings or that the model has learned generalisable semantic representations.

Table 21 presents the lexical features with the highest model-derived importance values for the Fake and Real class outputs. The reported values quantify the contribution of each feature to the corresponding model prediction within the evaluated corpus. Sev-

eral lexical features appear among the highest-ranked features for both classes, including "يرس", "كورو", and "صين", although their relative importance differs. Therefore, these values should be interpreted as context-dependent model associations rather than inherent semantic properties of the individual tokens.

For the Fake output, the largest reported model-derived importance values were associated with "كورو" (0.058), "يرس" (0.052), and "صين" (0.041). For the Real output, the largest reported values included "كورو" (0.056), "يرس" (0.053), and "صحه" (0.034). Because several tokens occur among the highest-ranked features for both outputs, the values should be interpreted only as model-specific associations whose magnitude depends on the fitted model and evaluated data. They do not establish that a token is intrinsically indicative of fake or real news, nor do they support factual, emotional, clinical, causal, or other semantic characterisations of the individual terms.

Table 21. Class-specific lexical features with the highest model-derived importance values.

Class	Rank	Feature	Importance
Fake	1	كورو	0.058
Fake	2	يرس	0.052
Fake	3	صين	0.041
Fake	4	امر	0.026
Fake	5	قتل	0.017
Real	1	كورو	0.056
Real	2	يرس	0.053
Real	3	صحه	0.034
Real	4	شفي	0.027
Real	5	مرض	0.018

Note: The English translations of the Arabic lexical features are: كورو (Corona/coronavirus), يرس (virus), صين (China), امر (order), قتل (killing/to kill), صحه (health), شفي (recovery/to recover), and مرض (disease/illness). The Arabic entries represent stemmed or normalised lexical forms generated during preprocessing; therefore, the English glosses indicate their closest contextual meaning rather than necessarily constituting literal translations of complete Arabic words or expressions. The importance values quantify the model-derived contribution of each lexical feature to the corresponding class output within the evaluated dataset. They should not be interpreted as indicating that an individual token is inherently fake, real, factual, emotional, or causally associated with misinformation.

Table 22 presents the top 20 globally important features extracted from the trained model using SHAP values. The results identify lexical features with relatively high model-derived importance values. These values represent statistical contributions to the model outputs within the evaluated corpus and should not be interpreted as inherent semantic or causal properties of the individual tokens. The importance values were normalised to percentage contributions to ensure interpretability and comparability. The percentage contribution of each feature was computed using Equation (29).

$$\text{Contribution}_i(\%) = \frac{|\phi_i|}{\sum_{j=1}^{p} |\phi_j|} \times 100, \tag{29}$$

where ϕ_i denotes the mean absolute SHAP value of feature i, and p denotes the total number of evaluated features.

To further evaluate the interpretability of the proposed model, a case study analysis was conducted on a set of representative news articles. Both fake and real news samples were selected to examine the classifier's decision-making process using local explanation techniques. For each instance, LIME was applied to identify the most influential words contributing to the final prediction. The results illustrate how individual lexical features

contributed to the selected predictions. These local contributions are specific to the examined instances and should not be interpreted as inherent indicators of fake or real news.

Table 22. Top 20 globally important lexical features based on mean absolute SHAP values.

Rank	Feature	Mean Absolute SHAP Value	Contribution (%)
1	صحه	0.029	16.2%
2	وضء	0.017	9.5%
3	بنء	0.015	8.4%
4	بن	0.010	5.6%
5	ايم	0.009	5.0%
6	بلز	0.008	4.5%
7	ندق	0.008	4.5%
8	تدخ	0.007	3.9%
9	حقق	0.006	3.3%
10	هل	0.006	3.3%
11	شفي	0.006	3.3%
12	دخن	0.006	3.3%
13	خلل	0.006	3.3%
14	ظهر	0.006	3.3%
15	كل	0.005	2.8%
16	تنب	0.005	2.8%
17	صحح	0.005	2.8%
18	وصل	0.005	2.8%
19	طلب	0.005	2.8%
20	خطر	0.004	2.2%

Note: The English glosses of the Arabic lexical features are provided for accessibility as follows: صحه (health), وضء (ablution), بنء (building/construction), بن (building), ايم (days), بلز (plasma), ندق (hotel), تدخ (enter), حقق (achieved), هل (whether/interrogative particle), شفي (recovery/to recover), دخن (smoking/smoke), خلل (defect/fault), ظهر (appearance/to appear), كل (all/every), تنب (prediction/to predict), صحح (correction/to correct), وصل (arrival/to reach), طلب (request/to request), and خطر (danger/risk). Because these entries are algorithmically generated stem/root forms following Arabic preprocessing, the English terms above are intended as approximate lexical glosses of the corresponding Arabic forms rather than as context-independent semantic interpretations. Contributions were calculated from the mean absolute SHAP values using Equation (29). The reported values represent model-derived feature-attribution magnitudes within the evaluated corpus only and should not be interpreted as assigning semantic, causal, factual, emotional, or class-intrinsic properties to the individual tokens.

The LIME framework was employed to generate local explanations for individual Arabic news documents. Specifically, three representative documents were selected from the testing dataset to analyse the decision-making behaviour of the proposed model. The reported explanations include representative correctly classified and misclassified testing instances selected to illustrate WDELM model behaviours. For each document, LIME generated three complementary explanation outputs:

- The original text with highlighted influential words;
- Local feature contribution scores;
- Predicted probabilities for fake and real classes.

The highlighted words identify terms assigned positive or negative local contributions by the LIME surrogate model for the selected prediction. The accompanying probability values report the WDELM output for that instance; they should not be interpreted as calibrated uncertainty estimates unless calibration is evaluated separately. Table 23 presents LIME-based explanations for three representative Arabic news documents. The examples report the local contribution values assigned to the displayed tokens for each selected prediction. In the first example, the largest-magnitude local contributions were assigned to “بن” and “كورو”. In the second example, the displayed tokens received contributions with opposing signs and comparatively small magnitudes. In the third example, “هل” and “خطر”

received the largest positive local contributions. These observations describe only the local surrogate explanations for the three selected documents and should not be generalised to the dataset as a whole.

Table 23. LIME-based local explanations for representative Arabic news documents.

Text	Actual	Predicted	Dominant Term	Contribution	Observed Local Explanation Pattern
حقق كتب خبر زمن ءلف راهيم بن سلق شره كورو كتب	Real	Fake	بن حقق كورو خبر كتب زمن ءلف سلق شره راهيم	−0.652 0.343 −0.300 0.161 0.088 0.053 0.027 0.026 0.021 0.004	The largest–magnitude contributions were assigned to بن and كورو.
يطل لاجازيتا وكل عمل ديبال ينف صبه ولو يرس كورو	Real	Fake	يطل ولو ينف كورو يرس لاجازيتا ديبال صبه وكل عمل	−0.234 −0.227 0.132 −0.105 0.047 0.042 −0.037 −0.027 0.005 −0.005	The displayed tokens received mixed positive and negative local contributions.
هل حمل لبس بله قدم ورب خطر	Fake	Real	هل خطر قدم لبس حمل ورب بله	0.347 0.341 0.034 0.023 0.012 0.007 0.000	The local explanation was dominated by هل and خطر.

Note: The final column summarises only the displayed LIME contribution values. It does not provide a semantic, causal, or factual interpretation of the Arabic tokens or of the documents. For clarity and accessibility, English lexical glosses of the Arabic tokens are provided below for each representative sample. Because the displayed Arabic forms were obtained after normalisation and ISRI stemming, some entries correspond to reduced stems rather than complete surface-form Arabic words. Accordingly, the English equivalents should be interpreted as contextual lexical glosses rather than as literal translations of grammatically complete sentences. **Sample 1:** ءلف (authored/composed), زمن (time), خبر (news/report), كتب (wrote/writing), حقق (verified/verification), كورو (Corona/coronavirus), شره (published/disseminated), سلق (context-dependent stemmed lexical form), بن (son of/by), and راهيم (Ibrahim; proper name), "The Truth About the Book Akhbar al-Zaman and Its Alleged Author, Ibrahim ibn Saluqiyyah: Did the Book Predict COVID-19?" **Sample 2:** عمل (work/activity), وكل (agent/representative), لاجازيتا (La Gazzetta; proper name), يطل (context-dependent stemmed lexical form), ولو (even if/although), صبه (infection/injury), ينف (denies/denial), ديبال (Dybala; proper name), كورو (Corona/coronavirus), and يرس (virus), "La Gazzetta: Dybala's agent denies that Paulo Dybala has contracted COVID-19." **Sample 3:** بله (context-dependent stemmed lexical form), لبس (wearing/clothing), حمل (carrying/to carry), هل (whether/interrogative particle), خطر (danger/risk), ورب (context-dependent stemmed lexical form), and قدم (foot/to present, depending on context), "Do second-hand clothes imported from Europe pose a risk of transmitting COVID-19?." The same English glosses apply to the corresponding Arabic entries in the ``Dominant Term" column. The "Observed Local Explanation Pattern" column summarises only the displayed LIME contribution values for the selected instances. These local explanations quantify model-derived feature contributions and should not be interpreted as assigning inherent semantic, causal, factual, emotional, or class-specific properties to the individual Arabic tokens or to the documents in which they occur.

Figure 10 illustrates the local explanation generated using the LIME framework for an individual Arabic news instance. Positive contribution scores indicate terms supporting the predicted category, whereas negative values correspond to words opposing the prediction. The visualisation reports the local contribution values assigned by LIME to the displayed lexical features for the selected instance.

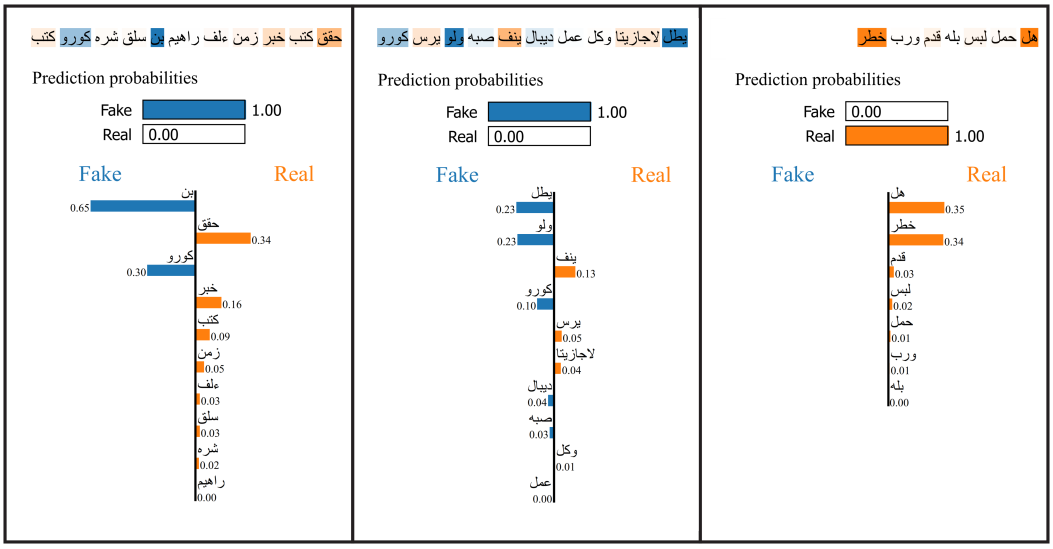

Figure 10. Local explanation generated using LIME for representative Arabic news instances. The visualisations report the prediction probabilities and the local contribution values assigned by LIME to the displayed lexical features. For clarity and accessibility, English lexical glosses of the Arabic tokens are provided as follows. **Left panel:** ءلف (authored/composed), زمن (time), خبر (news/report), كتب (wrote/writing), حقق (verified/verification), كورو (Corona/coronavirus), شره (published/disseminated), سلق (context-dependent stemmed lexical form), بن (son of/by), and راهيم (Ibrahim; proper name). **Middle panel:** عمل (work/activity), وكل (agent/representative), لاجازيتا (La Gazzetta; proper name), بطل (context-dependent stemmed lexical form), ولو (even if/although), صبه (infection/injury), ينف (denies/denial), ديبال (Dybala; proper name), كورو (Corona/coronavirus), and يرس (virus). **Right panel:** بله (context-dependent stemmed lexical form), لبس (wearing/clothing), حمل (carrying/to carry), هل (whether/interrogative particle), خطر (danger/risk), ورب (context-dependent stemmed lexical form), and قدم (foot/to present, depending on context). Because the displayed Arabic forms were obtained after normalisation and stemming, the English equivalents are intended as contextual lexical glosses rather than literal translations of complete Arabic expressions.

4.9. Discussion

The experimental results indicate that WDELM achieved the highest average performance among the evaluated methods under the adopted internal validation protocol. Its distance-aware adaptive weighting mechanism combines the posterior outputs of heterogeneous classifiers while assigning lower influence to learners whose prediction patterns show greater disagreement with the remaining ensemble members. Although several baseline methods achieved competitive results, WDELM remained statistically comparable with ET, ANN, RF, and LR across several metrics. These findings should therefore be interpreted as evidence of competitive internal performance rather than universal superiority or external generalisability.

The ablation analysis clarifies that final weight normalisation is not, by itself, a performance-enhancing component. Omitting normalisation rescales the ensemble output by the constant factor $m - 1$ and therefore preserves the sample ranking and ROC-AUC. When the mathematically equivalent threshold is used, the unnormalised and normalised formulations also produce identical binary predictions. The role of normalisation is consequently to preserve the probabilistic interpretation of the output and compatibility with the conventional threshold of 0.5.

The comparisons involving A1 and A2 should not be attributed solely to nonuniform classifier weighting because both variants employ majority voting over binary decisions and differ from the complete WDELM in several components. A7 was therefore introduced as the comparison: it uses the same six posterior-probability outputs, outer folds, fusion procedure, and threshold as WDELM, while replacing the fold-specific distance-aware weights with the uniform value $1/6$. Consequently, the A7–WDELM difference is

the appropriate quantity for assessing the incremental empirical contribution of the proposed weighting rule.

The ablation study provided additional evidence regarding the empirical contribution of the individual ensemble components. In the leave-one-classifier-out analysis, excluding XGBoost or RF produced the largest reductions in accuracy, with decreases of 2.483 percentage points in both cases. Removing AdaBoost caused the next-largest reduction of 2.207 percentage points. By contrast, excluding LGBM or LR reduced accuracy by 1.576 percentage points, while removing GB produced the smallest reduction of 1.261 percentage points.

The statistical analysis provided complementary evidence regarding performance differences among the evaluated machine learning methods. The Friedman test identified significant overall differences across the 12 classifiers under the matched ten-fold cross-validation protocol. Subsequent Wilcoxon signed-rank comparisons with Holm correction showed significant differences between WDELM and several baseline methods, whereas the differences from ET, ANN, RF, and LR were not statistically significant for most metrics. The Nemenyi analysis similarly placed WDELM, ET, and ANN within the leading statistical group. These findings should be interpreted within the adopted internal cross-validation protocol rather than as evidence of external generalisability or complete methodological superiority.

Another important finding concerns the comparison between conventional machine learning and DL approaches. Although DL models such as CNN, BiLSTM, GRU, and ResNet demonstrated competitive performance, they generally exhibited greater variability and lower stability than the proposed WDELM. Moreover, deep learning approaches required substantially greater computational resources and training time. In contrast, WDELM achieved good predictive performance while maintaining lower computational complexity and faster convergence behaviour. This finding indicates that carefully designed ensemble learning strategies can provide competitive predictive performance with lower computational complexity than the evaluated deep learning approaches, particularly for limited or domain-specific textual datasets.

The error analysis identified recurring observable characteristics among the examined misclassified samples, including shared lexical items across the two classes, short document length, and the presence of similar named entities or topic-specific terms. COVID-19-related items, for example, contained overlapping medical vocabulary in both classes. These observations are descriptive of the reviewed errors within the evaluated dataset and do not establish broader semantic or causal explanations for misclassification.

The LIME analysis provides local explanations of how individual lexical features contributed to selected WDELM predictions. The examined examples indicate that predictions can be influenced by a limited number of tokens, particularly in short or lexically ambiguous documents. These explanations improve the transparency of individual model outputs; however, they should not be interpreted as evidence that the identified words possess inherent semantic properties or that the model has learned generalised semantic representations.

From a practical perspective, the proposed WDELM demonstrates strong predictive performance, stable mean performance and competitive statistical-group placement under the adopted internal cross-validation protocol, computational efficiency, and model-output transparency. It therefore represents a research prototype that could support journalists, media organisations, fact-checkers, and policymakers by prioritising potentially suspicious news articles for further human verification. The dependence among the ten folds, the absence of pairwise effect sizes, and the lack of external validation nevertheless limit the inferential scope of these findings. SHAP and LIME expose model-derived feature

contributions for aggregate and selected local predictions, but these explanations should complement rather than replace professional judgement and should not be interpreted as semantic, causal, or factual validation of the highlighted terms. Furthermore, the proposed WDELM should be regarded as a research prototype rather than a deployment-ready misinformation detection system. Within a potential human-in-the-loop workflow, its probability estimates and local explanations could be used to prioritize potentially suspicious Arabic news items for subsequent review by professional fact-checkers. However, the present evaluation does not assess operational latency, domain shift, dialect variation, event drift, adversarial manipulation, or multimodal misinformation. Therefore, any practical use would require additional external validation, threshold calibration, monitoring, and expert oversight.

The experimental comparison demonstrates that the proposed WDELM remains competitive even when compared with recent pretrained Arabic transformer models. While transformer architectures benefit from contextual language representations learned from large-scale corpora, the proposed WDELM effectively exploits the complementary strengths of diverse machine learning classifiers through adaptive probability-based weighting. These findings suggest that ensemble diversity and distance-aware adaptive weighting remain effective strategies for Arabic fake news detection.

Future work will investigate the integration of transformer-based language models, including AraBERT and other pretrained Arabic models, into the WDELM framework. Additional research will examine multimodal misinformation by incorporating textual, visual, and social-network information. The proposed method will also be compared with alternative aggregation strategies, including hard voting, soft voting, stacking, and diversity-aware ensembles. Further experiments will evaluate inference latency, memory requirements, calibration at different operating thresholds, topic- and dialect-specific errors, domain adaptation, adversarial robustness, and integration into human-in-the-loop fact-checking workflows.

The findings indicate that the proposed WDELM achieved the highest average performance among the evaluated methods under the adopted experimental protocol while remaining statistically comparable with several leading baselines. The framework also provides a transparent probability-fusion mechanism and supports model-level interpretation through SHAP and LIME. However, the results are limited to the evaluated dataset and internal validation procedure and should not be interpreted as evidence of universal superiority, reduced overfitting, or generalisation across independent domains, dialects, and datasets.

4.10. Limitations

Despite the promising results achieved in this study, several limitations should be acknowledged.

- The original dataset records did not retain detailed acquisition, source, dialect, or platform-specific verification information, and only exact duplicate removal was applied. Future releases will preserve complete metadata and apply TF–IDF cosine-similarity screening with a threshold of 0.90 before dataset partitioning.
- The dataset size is relatively moderate and may not fully capture the diversity and complexity of real-world Arabic fake news. Expanding the dataset to include larger and more diverse sources would enable a more comprehensive evaluation of performance across topics, dialects, time periods, and news domains.
- The proposed method relies primarily on textual features, without incorporating additional modalities such as images, videos, or user interaction data. Since fake news

often involves multimodal content, integrating such information could enhance detection performance.

- The weighting mechanism assigns greater influence to classifiers whose posterior probability patterns show stronger agreement with the remaining ensemble members. However, agreement does not necessarily imply correctness, because several classifiers may produce similar but incorrect predictions. This limitation may reduce the effectiveness of consensus-based weighting in systematically ambiguous cases.
- The statistical comparisons were based on ten matched folds obtained from one dataset. Because the corresponding training partitions overlap, the fold-level metric estimates are correlated rather than independent. The reported intervals are therefore presented as descriptive fold-based intervals and not as conventional population-level confidence intervals. Pairwise effect sizes are reported together with the Wilcoxon statistics in the supplementary replication package; nevertheless, their interpretation remains limited to the adopted internal cross-validation protocol. Future work should additionally employ repeated cross-validation, appropriately corrected resampling procedures, and independent external datasets to obtain stronger inferential uncertainty estimates.
- The proposed model was evaluated only for binary classification on a single Arabic textual dataset under controlled experimental conditions. Its behaviour under multi-class settings, domain shift, dialect variation, evolving events, adversarial paraphrasing, multimodal misinformation, and real-time deployment constraints remains to be investigated.

5. Conclusions

This study introduced a WDELM framework for Arabic fake news detection. The proposed method combines posterior probability estimates from heterogeneous base classifiers using normalised cosine-distance-based weights derived from inter-classifier agreement in the prediction space. Under stratified ten-fold cross-validation, WDELM achieved a mean accuracy of $92.120 \pm 1.097\%$ and a mean ROC-AUC of $97.125 \pm 0.686\%$. Based on the concatenated predictions from the ten separately evaluated outer-validation folds, the framework achieved an F1-score of 91.357% for the Fake class, an F1-score of 92.759% for the Real class, and a pooled macro-F1 score of 92.058%. The method also obtained higher aggregate performance than the evaluated deep learning and pretrained Arabic transformer models under the common stratified 70–30% train–test protocol. The findings indicate that distance-aware probabilistic fusion can provide an effective and transparent mechanism for combining heterogeneous classifiers within the evaluated Arabic fake news dataset. However, the study was limited to a moderate-sized binary textual dataset and did not assess external datasets, multimodal misinformation, domain shift, dialect variation, adversarial manipulation, operational latency, memory consumption, or real-time deployment. Future work will extend the framework to larger and more diverse Arabic datasets, multi-class and multimodal settings, and transformer-based ensemble configurations. Additional evaluation will examine calibration, computational requirements, robustness under domain and temporal shifts, and integration into human-in-the-loop fact-checking workflows. Accordingly, WDELM should be regarded as a research prototype intended to support, rather than replace, professional fact-checking.

Author Contributions: Conceptualization, D.H.A. and M.F.M.; methodology, D.H.A.; software, D.H.A. and S.A.; validation, D.H.A., M.F.M., and L.K.T.; formal analysis, D.H.A.; investigation, D.H.A., M.F.M., L.K.T., W.K., S.A., and A.H.; resources, M.F.M.; data curation, D.H.A.; writing—original draft preparation, D.H.A.; writing—review and editing, M.F.M., L.K.T., W.K., S.A., and A.H.; visualisation,

D.H.A., Sam Ansari; supervision, M.F.M. and L.K.T.; project administration, M.F.M.; funding acquisition, L.K.T. All authors have read and agreed to the published version of the manuscript.

Funding: This research received no external funding.

Data Availability Statement: The final dataset supporting the findings of this study is publicly available through the Harvard Dataverse repository at https://doi.org/10.7910/DVN/A1JKYT. The repository contains the deposited dataset and its accompanying documentation.

Acknowledgments: The authors acknowledge the use of Grammarly v.1.2.271.1909 as a writing assistance tool to improve the clarity, grammar, and overall quality of this manuscript. All content and scientific contributions remain the responsibility of the authors.

Conflicts of Interest: The authors declare no conflicts of interest.

References

1. Radcliffe, D.; Bruni, P. State of Social Media, Middle East: 2018. *SSRN Electron. J.* **2019**, 1–40. . [CrossRef]
2. Radcliffe, D.; Yuan, H.; Yadav, N. How the Middle East Used Social Media in 2024. 2026. Available online: https://papers.ssrn.com/sol3/papers.cfm?abstract_id=6525739 (accessed on 27 July 2026).
3. Tabassum, A.; Patil, R.R. A survey on text pre-processing & feature extraction techniques in natural language processing. *Int. Res. J. Eng. Technol.* **2020**, *7*, 4864–4867.
4. Tufchi, S.; Yadav, A.; Ahmed, T. A comprehensive survey of multimodal fake news detection techniques: Advances, challenges, and opportunities. *Int. J. Multimed. Inf. Retr.* **2023**, *12*, 28. [CrossRef]
5. Zhou, X.; Zafarani, R. A survey of fake news: Fundamental theories, detection methods, and opportunities. *ACM Comput. Surv.* **2020**, *53*, 109.
6. Alghamdi, J.; Luo, S.; Lin, Y. A comprehensive survey on machine learning approaches for fake news detection. *Multimed. Tools Appl.* **2024**, *83*, 51009–51067. [CrossRef]
7. Othman, N.A.; Elzanfaly, D.S.; Elhawary, M.M.M. Arabic fake news detection using deep learning. *IEEE Access* **2024**, *12*, 122363–122376. [CrossRef]
8. Albalawi, Y.; Buckley, J.; Nikolov, N.S. Investigating the impact of pre-processing techniques and pre-trained word embeddings in detecting Arabic health information. *J. Big Data* **2021**, *8*, 95. [CrossRef] [PubMed]
9. Drungilas, V.; Vaičiukynas, E.; Ablonskis, L.; Čeponienė, L. Shapley values as a strategy for ensemble weights estimation. *Appl. Sci.* **2023**, *13*, 7010. [CrossRef]
10. Palanivinayagam, A.; El-Bayeh, C.Z.; Damaševičius, R. Twenty years of machine-learning-based text classification: A systematic review. *Algorithms* **2023**, *16*, 236. [CrossRef]
11. Sun, Y.; Shao, H.; Zhang, B. Ensemble based on accuracy and diversity weighting for evolving data streams. *Int. Arab. J. Inf. Technol.* **2022**, *19*, 90–96. [CrossRef] [PubMed]
12. Patil, D.R. Boosting ChatGPT sentiment classification with stacking and majority voting on skewed data. *Vietnam J. Comput. Sci.* **2026**, *13*, 267–299. [CrossRef]
13. Wu, S.; Li, J.; Ding, W. A geometric framework for multiclass ensemble classifiers. *Mach. Learn.* **2023**, *112*, 4929–4958. [CrossRef]
14. Chen, S.; Luc, N.M. RRMSE Voting Regressor: A weighting function based improvement to ensemble regression. *arXiv* **2022**, arXiv:2207.04837.
15. De Bock, K.W.; Bogaert, M.; du Jardin, P. Ensemble learning for operations research and business analytics. *Ann. Oper. Res.* **2025**, *353*, 419–448. [CrossRef]
16. Sagi, O.; Rokach, L. Ensemble learning: A survey. *Wiley Interdiscip. Rev. Data Min. Knowl. Discov.* **2018**, *8*, e1249. [CrossRef]
17. Nakach, F.Z.; Idri, A.; Goceri, E. A comprehensive investigation of multimodal deep learning fusion strategies for breast cancer classification. *Artif. Intell. Rev.* **2024**, *57*, 327. [CrossRef]
18. Rastogi, S.; Bansal, D. A review on fake news detection 3T's: Typology, time of detection, taxonomies. *Int. J. Inf. Secur.* **2023**, *22*, 177–212. [CrossRef] [PubMed]
19. Alazab, M.; Awajan, A.; Alazab, A.; Alhyari, A.; Saadeh, R. Fake-news detection system using machine-learning algorithms for Arabic-language content. *J. Theor. Appl. Inf. Technol.* **2022**, *100*, 5056–5069.
20. Alkhair, M.; Meftouh, K.; Smaïli, K.; Othman, N. An Arabic corpus of fake news: Collection, analysis and classification. In *Arabic Language Processing: From Theory to Practice—7th International Conference, ICALP 2019, Nancy, France, 16–17 October 2019, Proceedings*; Springer: Cham, Switzerland, 2019; pp. 292–302.
21. Himdi, H.; Weir, G.; Assiri, F.; Al-Barhamtoshy, H. Arabic fake news detection based on textual analysis. *Arab. J. Sci. Eng.* **2022**, *47*, 10453–10469. [CrossRef]

22. Khanam, Z.; Alwasel, B.; Sirafi, H.; Rashid, M. Fake news detection using machine learning approaches. *IOP Conf. Ser. Mater. Sci. Eng.* **2021**, *1099*, 012040. [CrossRef]
23. Alkadri, A.M.; Elkorany, A.; Ahmed, C. Enhancing detection of Arabic social spam using data augmentation and machine learning. *Appl. Sci.* **2022**, *12*, 11388. [CrossRef]
24. Thaher, T.; Saheb, M.; Turabieh, H.; Chantar, H. Intelligent detection of false information in Arabic tweets utilizing hybrid Harris Hawks based feature selection. *Symmetry* **2021**, *13*, 556. [CrossRef]
25. Roshan, R.; Bhacho, I.A.; Zai, S. Comparative analysis of TF–IDF and hashing vectorizer for fake news detection. *Eng. Proc.* **2023**, *46*, 5. [CrossRef]
26. Mahlous, A.R.; Al-Laith, A. Fake news detection in Arabic tweets during the COVID-19 pandemic. *Int. J. Adv. Comput. Sci. Appl.* **2021**, *12*, 778–788. [CrossRef]
27. Abualigah, L.; Al-Ajlouni, Y.Y.; Daoud, M.S.; Altalhi, M.; Migdady, H. Fake news detection using recurrent neural network based on bidirectional LSTM and GloVe. *Soc. Netw. Anal. Min.* **2024**, *14*, 40. [CrossRef]
28. Harrag, F.; Djahli, M.K. Arabic fake news detection: A fact checking based deep learning approach. *Trans. Asian Low-Resour. Lang. Inf. Process.* **2022**, *21*, 75. [CrossRef]
29. Fouad, K.M.; Sabbeh, S.F.; Medhat, W. Arabic fake news detection using deep learning. *Comput. Mater. Contin.* **2022**, *71*, 3647–3665. [CrossRef]
30. Antoun, W.; Baly, F.; Hajj, H. AraBERT: Transformer-based Model for Arabic Language Understanding. In *Proceedings of the 4th Workshop on Open-Source Arabic Corpora and Processing Tools, with a Shared Task on Offensive Language Detection*; Al-Khalifa, H., Magdy, W., Darwish, K., Elsayed, T., Mubarak, H., Eds.; European Language Resources Association: Paris, France, 2020; pp. 9–15.
31. Abdul-Mageed, M.; Elmadany, A.; Nagoudi, E.M.B. ARBERT & MARBERT: Deep Bidirectional Transformers for Arabic. In *Proceedings of the 59th Annual Meeting of the Association for Computational Linguistics and the 11th International Joint Conference on Natural Language Processing (Volume 1: Long Papers)*; Zong, C., Xia, F., Li, W., Navigli, R., Eds.; Association for Computational Linguistics: Stroudsburg, PA, USA, 2021; pp. 7088–7105. [CrossRef]
32. Al-Dabet, S.; ElMassry, A.; Alomar, B.; Alshamsi, A. Transformer-based Arabic Offensive Speech Detection. In Proceedings of the 2023 International Conference on Emerging Smart Computing and Informatics (ESCI), Pune, India, 1–3 March 2023; pp. 1–6. [CrossRef]
33. Abdelali, A.; Hassan, S.; Mubarak, H.; Darwish, K.; Samih, Y. Pre-Training BERT on Arabic Tweets: Practical Considerations. *arXiv* **2021**, arXiv:2102.10684.
34. Alrashidi, B.; Jamal, A.; Alkhathlan, A. Abusive Content Detection in Arabic Tweets Using Multi-Task Learning and Transformer-Based Models. *Appl. Sci.* **2023**, *13*, 5825. [CrossRef]
35. Antoun, W.; Baly, F.; Hajj, H. AraELECTRA: Pre-Training Text Discriminators for Arabic Language Understanding. In *Proceedings of the Sixth Arabic Natural Language Processing Workshop*; Habash, N., Bouamor, H., Hajj, H., Magdy, W., Zaghouani, W., Bougares, F., Tomeh, N., Abu Farha, I., Touileb, S., Eds.; Association for Computational Linguistics: Stroudsburg, PA, USA, 2021; pp. 191–195.
36. Alyoubi, S.; Kalkatawi, M.; Abukhodair, F. The detection of fake news in Arabic tweets using deep learning. *Appl. Sci.* **2023**, *13*, 8209. [CrossRef]
37. Al-Yahya, M.; Al-Khalifa, H.; Al-Baity, H.; AlSaeed, D.; Essam, A. Arabic fake news detection: Comparative study of neural networks and transformer-based approaches. *Complexity* **2021**, *2021*, 5516945. [CrossRef]
38. Elsaeed, E.; Ouda, O.; Elmogy, M.M.; Atwan, A.; El-Daydamony, E. Detecting fake news in social media using voting classifier. *IEEE Access* **2021**, *9*, 161909–161925. [CrossRef]
39. Nassif, A.B.; Elnagar, A.; Elgendy, O.; Afadar, Y. Arabic fake news detection based on deep contextualized embedding models. *Neural Comput. Appl.* **2022**, *34*, 16019–16032. [CrossRef] [PubMed]
40. Wotaifi, T.A.; Dhannoon, B.N. An effective hybrid deep neural network for Arabic fake news detection. *Baghdad Sci. J.* **2023**, *20*, 20. [CrossRef]
41. Abd, D.H.; Mahdi, M.F.; Jassim, M.A.; Hussain, A. Arabic fake news detection using ensemble technique. In Proceedings of the 2023 16th International Conference on Developments in eSystems Engineering (DeSE), Istanbul, Türkiye, 8–20 December 2023; pp. 292–297.
42. Touahri, I.; Mazroui, A. Survey of machine learning techniques for Arabic fake news detection. *Artif. Intell. Rev.* **2024**, *57*, 157. [CrossRef]
43. Almandouh, M.E.; Alrahmawy, M.F.; Eisa, M.; Elhoseny, M.; Tolba, A.S. Ensemble based high performance deep learning models for fake news detection. *Sci. Rep.* **2024**, *14*, 26591. [CrossRef] [PubMed]
44. Shehata, A.M.K.; Al-Suqri, M.N.; Osman, N.E.M.E.; Hamad, F.; Alhusaini, Y.N.; Mahfouz, A. ArabFake: A multitask deep learning framework for Arabic fake news detection. *IEEE Access* **2024**, *12*, 191345–191360. [CrossRef]

45. Syarief, M.G.; Kurahman, O.T.; Huda, A.F.; Darmalaksana, W. Improving Arabic stemmer: ISRI stemmer. In Proceedings of the 2019 IEEE 5th International Conference on Wireless and Telematics (ICWT), Yogyakarta, Indonesia, 25–26 July 2019; pp. 1–4.
46. Lichouri, M.; Abbas, M.; Lounnas, K.; Benaziz, B.; Zitouni, A. Arabic dialect identification based on a weighted concatenation of TF-IDF features. In *Proceedings of the Sixth Arabic Natural Language Processing Workshop*; Association for Computational Linguistics: Stroudsburg, PA, USA, 2021; pp. 282–286.
47. Minaee, S.; Kalchbrenner, N.; Cambria, E.; Nikzad, N.; Chenaghlu, M.; Gao, J. Deep learning-based text classification: A comprehensive review. *ACM Comput. Surv.* **2021**, *54*, 62.
48. Albtoush, E.S.; Gan, K.H.; Alrababa, S.A.A. Fake News Detection: State-of-the-Art Review and Advances with Attention to Arabic Language Aspects. *PeerJ Comput. Sci.* **2025**, *11*, e2693. [CrossRef] [PubMed]
49. Murshed, B.A.H.; Mallappa, S.; Abawajy, J.; Saif, M.A.N.; Al-ariki, H.D.E.; Abdulwahab, H.M. Short text topic modelling approaches in the context of big data: Taxonomy, survey, and analysis. *Artif. Intell. Rev.* **2023**, *56*, 5133–5260. [CrossRef] [PubMed]
50. Gasparetto, A.; Marcuzzo, M.; Zangari, A.; Albarelli, A. A survey on text classification algorithms: From text to predictions. *Information* **2022**, *13*, 83. [CrossRef]
51. Das, M.; Selvakumar, K.; Alphonse, P.J.A. A Comparative Study on TF-IDF Feature Weighting Method and Its Analysis Using Unstructured Dataset. *arXiv* **2023**, arXiv:2308.04037.
52. Da Costa, L.S.; Oliveira, I.L.; Fileto, R. Text classification using embeddings: A survey. *Knowl. Inf. Syst.* **2023**, *65*, 2761–2803. [CrossRef]