Fake News Detection in Palestinian News Using Machine Learning

Walid K. W. Alsafadi

Ahmed I. F. Alkhateeb

University College of Applied Sciences

DSAI 3308: Natural Language Processing

Dr. Tareq Altalmas

July 2025

Abstract

This project presents a complete machine learning pipeline to detect fake news in Arabic news articles about Palestine. 5,323 news articles were collected from over 20 sources between 2023 and 2025 by engaging students, web scraping, and Telegram APIs. Articles were annotated with credible news sources for authentic content (e.g., Al Jazeera) and verified fact-checking sites for false content (e.g., Misbar, Tibyan).

Massive exploratory data analysis revealed critical problems like platform bias, class imbalance, and lexical difference between authentic and fake content. Two preprocessing techniques—minimum and aggressive cleaning—were employed to see how it would impact model performance. TF-IDF was applied as the feature extraction technique, and four models were evaluated: Logistic Regression, Multinomial Naive Bayes, Random Forest, and XGBoost.

XGBoost was the best with a cross-validated F1-score of 0.842. The model was made available as an interactive web application in Arabic. Real-world deployment potential has been demonstrated, and future upgrades can be made using deep learning models, better UI, and software engineering practices.

Keywords*: Arabic NLP, fake news detection, TF-IDF, XGBoost, Palestine, machine learning.*

1. Introduction

In last few years, the spread of fake news increased and become a major challenge, especially in political region. With the rise of social media platforms and many unverified information. It makes it more difficult to classify it especially news written in slang. The need of AI has become more urgent that ever. Misinformation in conflicts can lead to widely spread and waste lifes, dangoures, and emotionally damage people, and real world harm.

This project focuses on deliver machine learning-based application to let users easily check if the news are real or fake with a confidence score for Palestinian related articles. The model classify articles based on different factors such as textual features, and linguistic patterns. This system is designed to support readers, journalists, and fact-checkers.

The projects follows an end-to-end pipeline starting from collecting data, exploratory analysis, preprocessing, feature engineering, model selection, evaluation, and deployment. The final model is integrated with user-friendly Arabic interface hosted on streamlit. Designed to be user for non-techonlogy familiar users to let them easily check articles with a single click.

2. Dataset Description and Analysis

The section introduces the dataset used in the project, including its organization, label distribution, platform diversity, text characteristics, and data cleaning processes. It also contains our judgment on their quality and limitations.

2.1. Dataset Overview and Structure

This project's dataset contains 5,323 news articles in Arabic, all of which are related to Palestine's activities. The data was collected in a time span from June 2023 until July 2025, from a combination of Telegram APIs, web scraping, and manual entry. Articles were labeled real or false based on the credibility source.

Each article is represented by several fields:

- **title**: Title of the news headline
- **content**: Article body
- **platform**: Publishing outlet or source name (e.g., Al Jazeera, Misbar)
- **date**: Publication date in YYYY-MM-DD HH:MM:SS format
- **label**: Real or Fake ground truth label

This setup enabled both text-level modeling and metadata-level analysis (e.g., platform bias, temporal patterns).

2.2. Label Distribution

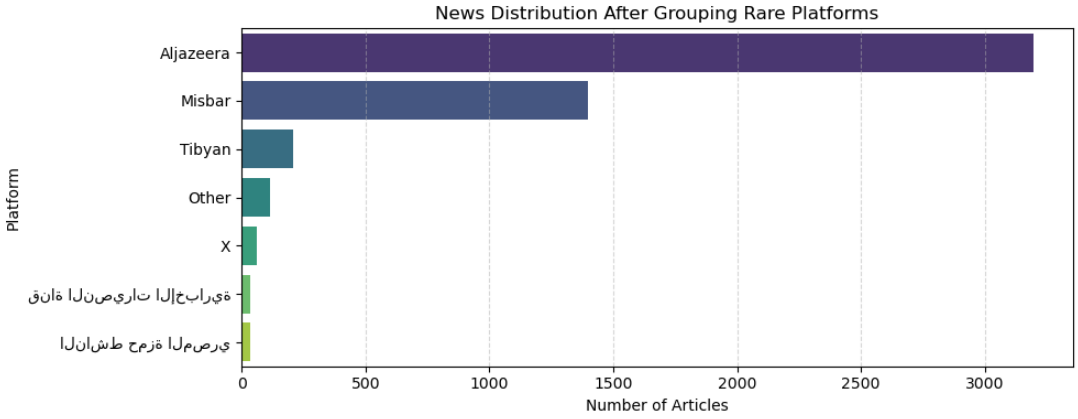
One of the earliest observations was that the dataset had a moderate class bias. Roughly 72.3% of the samples are labeled as real, and 27.7% as fake. While not as extreme, the bias is risky to overlook since it can bias the model toward the dominant class. This reality immediately influenced the selection of evaluation metrics (favoring F1-score over accuracy) and construction of preprocessing and training methods.

2.3. Platform Analysis

The articles were received from over 20 different platforms. Actual news samples were mostly collected from trusted sites such as Al Jazeera, while false news examples were received from fact-checking sites such as Misbar and Tibyan, and anonymous or casual sources on Telegram and social media. Figure 1 illustrates Al Jazeera dominates the dataset with over 3,000 articles, followed by Misbar and Tibyan.

Observe that Misbar and Tibyan themselves are not sources of fabricated news. Instead, they capture and debunk common misinformation. Therefore, the samples identified as fabricated were obtained from verified fact-checks, which help in the simulation of realistic fabricated content for training purposes.

***Figure 1**: Total number of news articles by platform after grouping rare sources under "Other".*

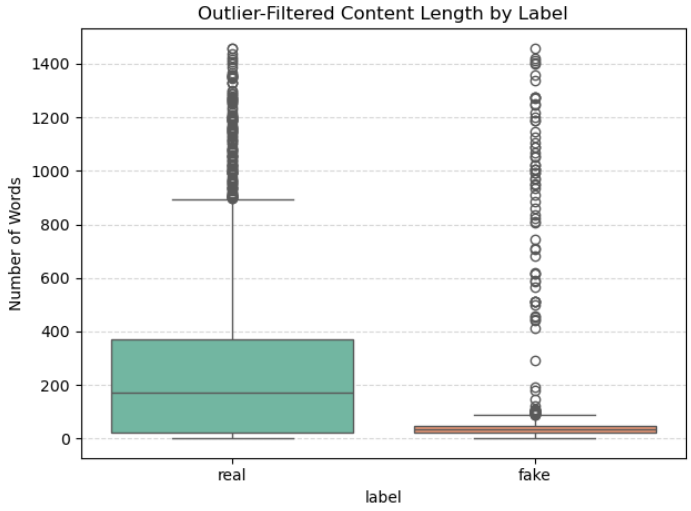


2.4. Text Length Patterns

Text length, that is, the number of words used in title and content, differed widely across authentic and fake news. Authentic posts are longer and provide more information, while spurious posts are shorter and less formal.

To give this a rough number, we removed outliers through the IQR method and then counted words. As shown in Figure 2, genuine news articles had an average content length of ~273 words, whereas spurious ones had an average of ~58 words — a substantial difference used as a later predictive feature.

***Figure 2**: illustrates content length distribution by label.*



2.5. Pre-Modeling Cleanup Steps

To ensure data consistency, we employed several cleaning steps:

- Dropped ~580 rows with duplicate or irrelevant records
- Deleted previous articles (prior to 2023)
- Normalized column names.
- Combined rare platforms into an "Other" category
- Created additional features: title_length, content_length, platform_encoded
- Standardized the date column to datetime format

These steps were formalized into reusable Python scripts to ensure reproducibility.

2.6. Final Cleaned Dataset

After preprocessing, the final cleaned dataset contained 4,743 articles. Both the minimal and aggressive versions of the text were kept available for experimentation with different model pipelines. The dataset was dumped in CSV format and saved as input for TF-IDF feature extraction and model training.

2.7. Opinion on Dataset Quality and Limitations

In my view, the dataset is a realistic and practical foundation upon which to build Arabic fake news detection systems. Fact-checking sites' labeling of fake samples lent greater legitimacy, while sources' diversity introduced essential differences in content quality and writing style.

However, there are certain limitations. Platform bias (Al Jazeera = true, Misbar = false) can cause models to overfit on source-based features rather than linguistic patterns. Moreover, since not all articles gathered by students were manually labelled, the label consistency may vary

a little bit. However, the final dataset is big enough, well-balanced for binary classification, and extremely relevant to the real-world issue of misinformation in the Palestinian context.

3. Data Cleaning and Preprocessing

This section describes the data preparation procedure used to preprocess and normalize raw Arabic text to be used as usable machine learning inputs. It also outlines other features designed from metadata for improving model performance.

3.1. Strategies for Text Cleaning

Two levels of text cleaning were performed in preparation for the model: minimal and aggressive. The techniques were utilized to determine the impact of preprocessing granularity on model performance.

3.1.1. Minimal Cleaning

The minimal version preserved more of the word structure of the original and included the following steps:

- Normalized characters (e.g., normalized "ٕ" ,"ٖ" ,"ٔ" to "ا")
- Filtered out non-Arabic chars and unnecessary whitespace

This variant was designed to keep as much linguistic information as possible, and this could be beneficial to models that can handle noisy text.

3.1.2. Aggressive Cleanining

This pipeline employed more aggressive transformations to remove linguistic noise and irrelevant tokens. It involved:

- All minimal operations, along with:
- Removed diacritics (tashkeel)
- Tokenized the text into individual words

- Removed Arabic stopwords using Taha Zerrouki stopwords list
- Removed short tokens (less than 2 characters)

Both were saved as text_clean_min and text_clean_agg, respectively, and were used standalone in different model pipelines for comparison.

3.2. Feature Engineering

In addition to text pre-processing, several metadata-based features were engineered so that the model can detect structural patterns higher than raw text. These are:

- title_length: Word count of article title
- content_length: Word count of article body
- platform_encoded: One-hot encoding or clustering of publication platforms

These features were applied for exploratory analysis and optionally added to model input to test their predictive strength.

3.3. Vectorization: TF-IDF

Term Frequency–Inverse Document Frequency (TF-IDF) vectorization was employed for transforming the cleaned text into numerical vectors. It determines the relative importance of each word in a document compared to its frequency within the entire collection.

TF-IDF down-weights very common words and up-weights context words. TF-IDF does a better job at detecting stylistic variation between genuine and fake news — especially in Arabic, where word repetition and morphology are extremely varied.

We employed:

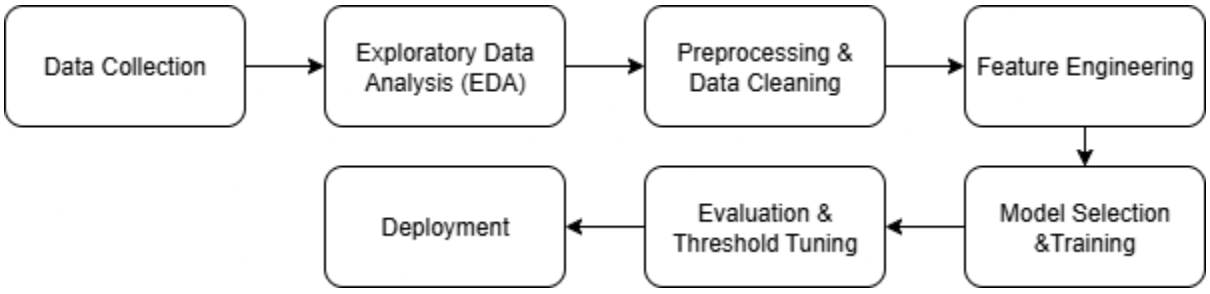
- ngram_range = (1, 2) to keep unigrams and bigrams
- Max features = 15,000 (for performance optimization)
- sublinear_tf = True to normalize term frequency

The generated sparse matrices were utilized as input features for all the classification models.

4. Methodology and Workflow

The project follows a complete end-to-end machine learning pipeline with seven primary steps (Figure 3). Each step was designed to be reproducible, modular, and experimentation-friendly. The workflow accepts raw Arabic news data and transforms it into an operational fake news detection system, exposed through a web-based Arabic interface.

***Figure 3**: End-to-end pipeline of the fake news detection system.*



4.1. Data Collection

Arabic news were collected in range 2023 to 2025 in collaboration between students, data was labeled as real or fake using validated sources such as Al Jazeera, Misbar, Tibyan. It was challenging to ensure a data validity especially for fake news so we relied on fact-checker website to ensure that fake news are a real-life. Data collection was done by various methods such as Telegram APIs, web scraping, and manual entries. At the end we merged our collected data which results a valid 5,323 records.

Labeling was performed according to trustworthy sources:

- Real news were validated using sites such as Al Jazeera
- Fake articles were identified using fact-checking websites like Misbar and Tibyan

This step ensured that the fake news samples were from actual misinformation in the real world, as opposed to artificially created examples.

4.2. Exploratory Data Analysis (EDA)

Before modeling, we performed extensive EDA to uncover patterns, biases, and issues in the dataset. Key points examined are class imbalance, platform distribution, writing style differences, and time-based trends.

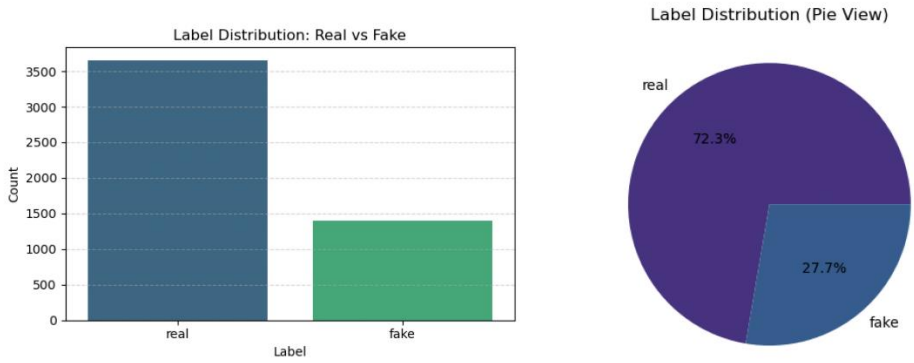
4.2.1. Label Distribution

As shown in **Figure 4**, the dataset is moderately imbalanced:

- ~72% of the articles are labeled as **real**
- ~28% are labeled as **fake**

This imbalance has implications for training, requiring weighted models or threshold tuning to avoid bias toward the majority class.

***Figure 4**: Label distribution in the dataset: (a) bar chart, (b) pie chart*



4.2.2. Platform Biases

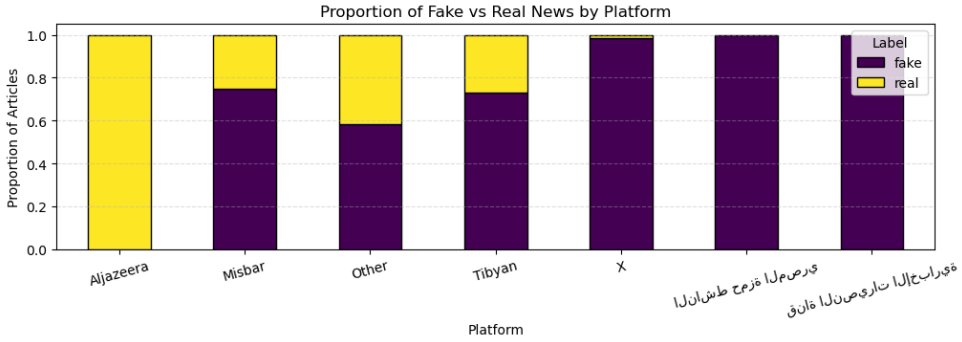
Certain platforms are strongly associated with either real or fake news. **Figure 5** shows that:

- **Al Jazeera** contributes almost exclusively real news

- **Misbar** and **Tibyan** contribute mostly fake-labeled articles
- Lesser-known sources (e.g., Telegram accounts, X) also skew toward fake

It's important to note that Misbar and Tibyan are **not** fake news platforms themselves, they are trusted fact-checkers whose documented findings were used to label misinformation.

***Figure 5**: Proportion of fake vs real news by platform*



4.2.3. Temporal Trends

We observed **temporal spikes** in article volume that align with key events. As shown in **Figure 6**, a significant rise in article frequency occurred following the **October 7, 2023** conflict escalation, peaking around December 2023.

In **Figure 7**, we compare the number of real vs fake articles over time. Although real news dominates overall, fake news **temporarily overtook** real news volume in **November 2023** and **April 2024.**

***Figure 6**: Monthly news volume with conflict event marker*

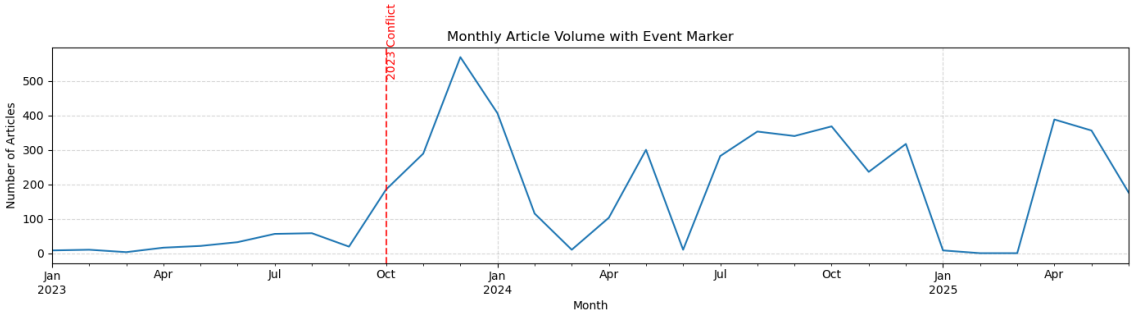
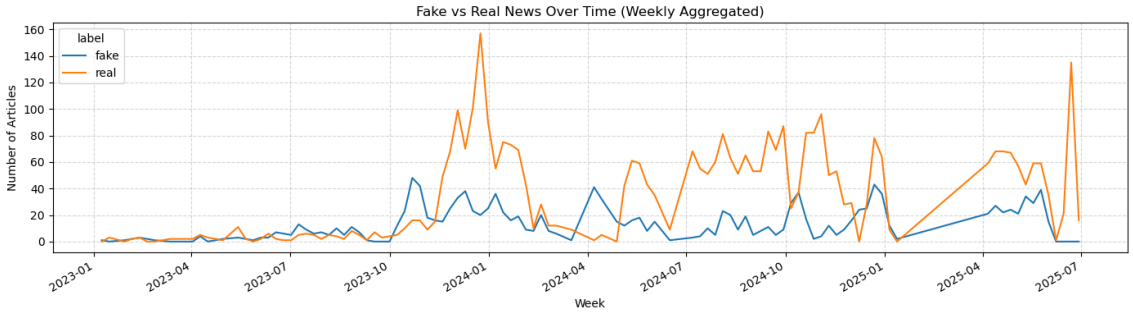


***Figure 7:** Weekly comparison of real vs fake news counts*



4.2.4. Lexical and word patterns

4.2.4.1 Most common words

We analyzed the most frequent words in each class. As shown in **Table 1** and **Table 2**, real news tends to contain military and political terms (e.g., "الجيش", "الاحتلال", "غزة"), while fake news often uses terms from social media or informal reporting (e.g., "حسابات", "فيديو", "التواصل").

***Table 1**: Most Frequent Words in **Real** News Articles*

Word	Count
(Gaza) غزة	7,293
(Occupation) الاحتلال	5,815
(Israeli) الإسرائيلي	5,650

إسرائيل (Isreal)	4,031
قطاع (Strip)	3,512
الإسرائيلية (Israeli)	2,953
الجيش (Army)	2,409
حماس (Hamas)	2,319
القوات (Forces)	2,280
الحرب (War)	2,219

***Table 2**: Most Frequent Words in **Fake** News Articles*

Word	Count
غزة (Gaza)	1,209
التواصل (Communication)	625
الإسرائيلي (Israeli)	596
قطاع (Strip)	589
فيديو (Video)	556
حسابات (Accounts)	542
مقطع (Clip)	532
خلال (Through)	518
أنه (its)	500
إسرائيل (Isreal)	471

Note. Words are presented in Arabic with English glosses for clarity.

We concluded that real news common words are more army related, while fake news are more social media related.

4.2.4.2. Word Cloud

In an attempt to visualize lexical differences, side-by-side word clouds were created after removing Arabic stopwords. As apparent in Figure 8, real news is centered around conflict and military entity-related words, while fake news is centered around social media mentions.

***Figure 8**: Word cloud comparison for real vs fake news articles*

4.3. Preprocessing and Cleaning

We employed general data cleaning and text-specific preprocessing. We first:

- Removed pre-2023 and irrelevant articles, and duplicate rows
- Renamed and standardized columns and dates
- Collapsed rare platforms into an "Other" category

Then, we cleaned the text through two separate pipelines:

- Minimal cleaning: Normalized Arabic letters, lowercased text, removed non-Arabic characters and extra whitespace
- Aggressive cleaning: All minimal steps, plus diacritic removal, Arabic stopword filtering, tokenization, and removal of short tokens

This two-pronged approach allowed us to compare model performance at different levels of text quality later.

4.4. Feature Engineering

We engineered extra features to supplement model inputs:

- title_length: Number of words in the title
- content_length: Number of words in the article body
- platform_encoded: Label encoding or grouping of source platforms

These features were added to supplement the main text data and allow models to capture patterns involving source credibility or article structure.

4.5. Model Selection and Training

We trained four classical classifiers:

- Logistic Regression (LR)
- Multinomial Naive Bayes (MNB)
- Random Forest (RF)
- XGBoost (XGB)

We trained every model on TF-IDF vectorized text. We utilized 5-fold cross-validation and assessed minimal and aggressively cleaned datasets separately.

XGBoost yielded the best F1 scores consistently, which can most probably be attributed to its ability for modeling non-linear relationships along with its proficiency in handling imbalanced data. Multinomial Naive Bayes performed worse, most probably due to its strict assumptions about word independence.

4.6. Evaluation and Deployment

We evaluated model performance as:

- Precision, Recall, F1-Score
- Cross-validation F1 scores
- Confusion matrix

As shown on Table 3, XGBoost yielded a cross-validated F1 score of 0.842 on the minimal-cleaned dataset, with Logistic Regression and Random Forest close behind. One the other hand, in Table 4, XGBoost again had outstanding performance with a cross-validated F1 score of 0.836 on aggressive-cleaned dataset. Models had an overall better performance on minimal cleaned data than aggressive one.

Table 3: Model evaluation on minimal cleaning dataset

Model	Accuracy	F1-score	CV F1-score	CV F1-score Std
LogisticRegression	0.907	0.846	0.836	0.027
RandomForest	0.901	0.829	0.826	0.027
XGBoost	0.888	0.816	0.842	0.028
MultinomialNB	0.861	0.729	0.760	0.026

Table 4: Model evaluation on aggressive cleaning dataset

Model	Accuracy	F1-score	CV F1-score	CV F1-score Std
LogisticRegression	0.913	0.849	0.836	0.029
RandomForest	0.905	0.842	0.835	0.026
XGBoost	0.892	0.822	0.836	0.027

MultinomialNB	0.867	0.742	0.763	0.028

5. Discussion

This discussion reflects on the modeling decisions, preprocessing solutions, and model performance in our construction of the Arabic fake news detection system.

5.1. TF-IDF Justification

For the translation of Arabic text into numerical inputs compatible with machine learning models, we used Term Frequency–Inverse Document Frequency (TF-IDF). TF-IDF places greater weight on words that appear frequently in a specific document but not as often in the entire dataset. This helped to identify more typical patterns between real and fake news stories than a straightforward Bag-of-Words (BoW) model.

Unlike BoW, TF-IDF reduces the influence of very frequent words and focuses on context-related keywords, which is critical in capturing subtle variation between authentic and simulated content, especially political news.

5.2. Dealing with Imbalanced Data

Class imbalance was also the biggest issue during training. About 72% of our dataset is actual news, and there is a lack of fake news. We tried to avoid oversampling/undersampling to maintain ourselves away from overfitting or data loss. Instead, we battled imbalance by:

- Class weighting in model parameters (e.g., class_weight="balanced" in Logistic Regression and Random Forest)
- Threshold tuning to improve sensitivity to the minority class
- For XGBoost, we used scale_pos_weight=1.8, which allows the model to learn more from minority fake samples

These approaches treated the two classes with equal importance without altering the natural distribution of data.

5.3. Best Model Analysis (XGB vs others)

The top performance was by XGBoost among all the models we tried, which yielded a cross-validated F1-score of 84.2%. Its gradient boosting nature allows it to learn complex non-linear relationships and accommodate imbalanced datasets naturally.

Logistic Regression and Random Forest also performed well with 83.6% and 82.6% respectively. Multinomial Naive Bayes did not perform well with strong feature independence assumptions and poor class imbalance management, which most likely limited its practical utility.

5.4. Limitations and Insights

- **Platform bias**: Another significant issue is platform-label correlation — most of the real news came from Al Jazeera, while Misbar and Tibyan dominated fake labels. This automatically may lead to models labeling as credible depending on the source, rather than content. Better balanced and diverse representation of platforms would help in generalization.
- **Shortage of deep learning models**: With time and resource limitations, we did not attempt deep learning architectures such as LSTM, CNN, or transformer-based models (e.g., BERT). These could potentially learn deeper patterns of context, especially in the case of Arabic text.
- **Limitations of TF-IDF**: Powerful as it is, TF-IDF does not consider word order and context. Future developments of this project should consider word embeddings such as Word2Vec, FastText, or AraBERT to achieve improved semantic understanding.

6. Conclusion and Future Work

In this project, we suggested an end-to-end Arabic fake news detection system for Palestinian news. We collected 5,323 articles from over 20 platforms between the years 2023 and 2025 through teamwork and student collaboration. Real articles were confirmed with reliable

sources like Al Jazeera, while fake news examples were taken from fact-checking websites like Misbar and Tibyan for realistic and credible examples of misinformation.

We performed thorough exploratory data analysis (EDA) to identify and correct issues such as duplicates, old records, missing values, and class imbalance. To enable the models to perform better, we performed consistent preprocessing, combined low-frequency platforms, feature-engineered new features, and removed outliers.

We used TF-IDF to vectorize text features and trained four classifiers. XGBoost performed best, followed by Random Forest. Evaluation metrics used were precision, recall, F1-score, and cross-validation, which we used instead of accuracy for better portrayal of real-world performance on imbalanced data.

For future work and develop this system, we will:

- Grow the dataset with more diverse and balanced samples
- Explore deep learning architectures such as LSTM, BERT, and CNN for better contextual understanding
- Develop a production-grade system using Next.js, API backends, and cloud databases
- Apply software engineering and MLOps principles for scalability, reproducibility, and continuous enhancement

With these enhancements, we will develop a competitive, real-world Arabic fake news detector for the Palestinian context, with higher accuracy and fewer false positives.