

---

# NTT Data ML Challenge 2025

## Portuguese Fake News Detection

Group 9 – Instituto Superior Técnico

---

**Daniel Martins Borges**

ist1109474

daniel.m.borges@tecnico.ulisboa.pt

**Duarte Pereira da Cruz**

ist1110181

duarte.cruz@tecnico.ulisboa.pt

## Abstract

This work addresses the "Merit Prize 2025-26" challenge on detecting misinformation in the *FakeNews-PT* dataset. Five classifiers are benchmarked: Decision Tree, Gaussian Naive Bayes, Logistic Regression (L1 and L2), and Multi-Layer Perceptrons (MLP). Four distinct preprocessing pipelines are evaluated to identify the optimal strategy for model training. To interpret the decision-making logic of the classifiers, LIME and permutation importance are utilized. Finally, unsupervised analysis is performed using K-Means and PCA to uncover and visualize thematic clusters within the news corpus. The implementation code is available at <https://github.com/DuarteCruz6/MLChallenge-2526>.

## 1 Introduction

The proliferation of digital misinformation poses a critical challenge to modern information ecosystems, undermining public trust and democratic processes. While automated detection systems have matured for English-language content, resources for other languages remain comparatively scarce. This gap is particularly evident in Portuguese media, necessitating the development of specialized tools capable of distinguishing legitimate reporting from fabricated narratives.

The primary objective of the "Merit Prize 2025-26" challenge is to build robust machine learning models for binary classification of news articles and to analyze their decision-making processes. This report benchmarks the performance of probabilistic, tree-based, and neural network architectures on the *FakeNews-PT* dataset.

## 2 Data Overview and Preprocessing

This study utilizes the *FakeNews-PT* dataset, a large-scale corpus designed for the detection of Portuguese fake news sourced from the `fake-news-pt-eu` repository. The dataset comprises over 60,000 news articles, partitioned into training, validation, and testing sets to ensure robust evaluation.

The data is structured into two primary fields:

- **Text:** The raw string content of the news article.
- **Label:** A binary integer target, where 0 represents fake news and 1 represents real news.

### 2.1 Preprocessing Pipeline

To prepare the raw text for feature extraction, a flexible pipeline was implemented to test the impact of punctuation and stop words on model performance. Four distinct types of cleaning were applied:

1. **No punctuation:** The text was normalized by removing all punctuation marks, converting all characters to lowercase, and stripping extra whitespaces.
2. **No punctuation and no Portuguese stop words:** In addition to the steps above (punctuation removal, lowercase conversion, whitespace stripping), common Portuguese stop words were filtered out using the NLTK library to focus on high-value semantic terms.
3. **With punctuation:** Punctuation marks were preserved to retain potential syntactic signals. The text was otherwise converted to lowercase and cleared of extra whitespaces.
4. **With punctuation and no Portuguese stop words:** Punctuation was preserved, but Portuguese stop words were removed. The text was also converted to lowercase and stripped of extra whitespaces.

It is important to note that no manual tokenization was performed during these steps. Instead, the cleaned text strings were passed directly to the TF-IDF vectorizer, which internally handles tokenization during feature construction.

### 3 Feature Extraction

To transform the unstructured textual data into a numerical format suitable for machine learning, the Term Frequency-Inverse Document Frequency (TF-IDF) representation was employed. This method is particularly appropriate for text classification tasks as it balances term frequency with information content. By assigning lower weights to words that appear in the majority of documents (and thus possess low discriminative power), TF-IDF prioritizes terms that are distinctive to specific articles, allowing the model to focus on the unique vocabulary that differentiates real news from fabricated content.

The vectorization process was applied directly to the preprocessed text strings. To maintain computational efficiency and reduce the risk of overfitting due to high dimensionality, the vocabulary size was constrained to the top 5,000 features ordered by term frequency.

To further refine the feature space and ensure robustness against noise, strict document frequency boundaries were imposed. A minimum document frequency (*min\_df*) of 10 was set to ignore extremely rare terms or typos that do not generalize well. Conversely, a maximum document frequency (*max\_df*) of 0.9 was applied to exclude corpus-specific stop words that appear in more than 90% of the documents. Finally, IDF smoothing was enabled to prevent division by zero and to provide a more stable weighting scheme for out-of-vocabulary words.

### 4 Model Training

Five distinct classifiers were evaluated to cover a range of learning paradigms, including tree-based, probabilistic, linear, and neural approaches. To ensure the statistical robustness of the results, all models were trained using 5-fold cross-validation. For each model, a systematic grid search was performed to tune key hyperparameters, selecting the optimal configuration based on the macro-F1 score.

While these models were trained across all four preprocessing pipelines defined in Section 2, the results presented in this section focus exclusively on the "With punctuation and no Portuguese stop words" dataset. As demonstrated in the subsequent chapter, this specific preprocessing strategy consistently yielded the highest performance metrics across the majority of classifiers.

#### 4.1 Decision Tree

The depth and split criteria were explored to balance the tree's complexity and its ability to generalize.

- **Hyperparameter Grid:**
  - `max_depth`: [10, 20, 50, None]
  - `min_samples_split`: [2, 5, 10]
- **Best Configuration:** `{max_depth: 50, min_samples_split: 2}`

## 4.2 Gaussian Naive Bayes

For the Gaussian Naive Bayes classifier, the variance smoothing parameter was tuned to address numerical stability and distribution assumptions.

- **Hyperparameter Grid:**
  - `var_smoothing`: [1e-9, 1e-8, 1e-7, 1e-6]
- **Best Configuration:** `{var_smoothing: 1e-06}`

## 4.3 Logistic Regression (L2 Regularization)

L2 (Ridge) regularization was applied to prevent overfitting by penalizing large coefficients, with tuning performed on the regularization strength ( $C$ ) and solver.

- **Hyperparameter Grid:**
  - `C`: [0.01, 0.1, 1, 10, 100]
  - `solver`: ['liblinear', 'saga']
- **Best Configuration:** `{C: 10, solver: 'saga'}`

## 4.4 Logistic Regression (L1 Regularization)

L1 (Lasso) regularization was tested to induce sparsity in the feature weights, potentially performing automatic feature selection.

- **Hyperparameter Grid:**
  - `C`: [0.01, 0.1, 1, 10]
  - `solver`: ['liblinear', 'saga']
- **Best Configuration:** `{C: 1, solver: 'saga'}`

## 4.5 Multi-Layer Perceptron (MLP)

A neural network was trained with up to two hidden layers, with optimization performed on the architecture, regularization alpha, and learning rate.

- **Hyperparameter Grid:**
  - `hidden_layer_sizes`: [(128,), (256,), (128, 64), (256, 128)]
  - `alpha`: [0.0001, 0.001, 0.01]
  - `learning_rate_init`: [0.0001, 0.001]
- **Best Configuration:** `{hidden_layer_sizes: (256, 128), alpha: 0.01, learning_rate_init: 0.001}`

# 5 Model Evaluation

This section presents a comparative analysis of model performance. The selection of the primary dataset is first substantiated, followed by a benchmark of all five classifiers on the held-out test set.

## 5.1 Dataset Selection

To determine the optimal preprocessing strategy, the performance metrics of all models were averaged across the four available datasets. As presented in Table 1, the dataset preprocessed with "With punctuation + No stopwords" achieved the highest scores across all key metrics, including accuracy and F1-macro. Consequently, all subsequent model comparisons in this report are based on this specific dataset.

Table 1: Average test metrics across the four preprocessing pipelines.

Dataset	Accuracy	Precision (Macro)	Recall (Macro)	F1 (Macro)
No Punctuation	0.8058	0.8073	0.8058	0.8055
No Punct. + No Stopwords	0.8855	0.8857	0.8855	0.8855
With Punctuation	0.8869	0.8869	0.8869	0.8869
<b>With Punct. + No Stopwords</b>	<b>0.8875</b>	<b>0.8876</b>	<b>0.8875</b>	<b>0.8875</b>

## 5.2 Model Comparison

Table 2 summarizes the performance of the five classifiers on the selected test set. The results indicate that the Multi-Layer Perceptron (MLP) outperforms all other models, achieving the highest Accuracy (0.9261) and F1-score (0.9261).

Table 2: Test set metrics for all classifiers on the "With punctuation + No stopwords" dataset.

Model	Accuracy	Precision (Macro)	Recall (Macro)	F1 (Macro)
Decision Tree	0.8589	0.8592	0.8588	0.8588
Gaussian NB	0.8516	0.8517	0.8516	0.8516
Logistic Regression (L2)	0.9000	0.9000	0.9000	0.9000
Logistic Regression (L1)	0.9009	0.9010	0.9009	0.9009
<b>MLP</b>	<b>0.9261</b>	<b>0.9262</b>	<b>0.9261</b>	<b>0.9261</b>

## 5.3 Best Classifier Analysis

Given its superior performance, the Multi-Layer Perceptron (MLP) was selected as the best classifier. Figure 1 displays the Receiver Operating Characteristic (ROC) curve for the MLP model. The curve demonstrates strong discriminative ability, with an Area Under the Curve (AUC) of 0.9779. This confirms the model's effectiveness in distinguishing between real and fake news articles with a high true positive rate while maintaining a low false positive rate.

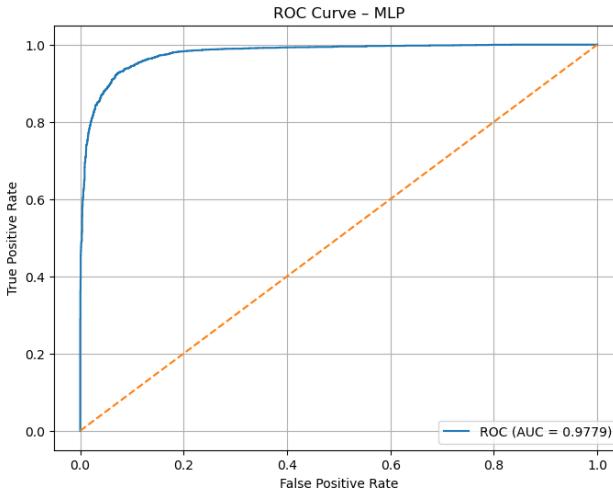


Figure 1: ROC Curve for the Multi-Layer Perceptron (MLP) classifier.

## 6 Model Interpretation

To move beyond black-box predictions and ensure transparency, interpretability methods were applied to inspect the internal logic of the trained models.

## 6.1 Feature Importance (Logistic Regression)

The coefficients of the L1-regularized Logistic Regression model were analyzed to identify the lexical features most indicative of each class. Since L1 regularization encourages sparsity, it effectively highlights the most salient terms driving the decision boundary. Figure 2 visualizes the top 10 words with the largest positive and negative weights, demonstrating which terms strongly influence the model's classification towards "Real" or "Fake" news.

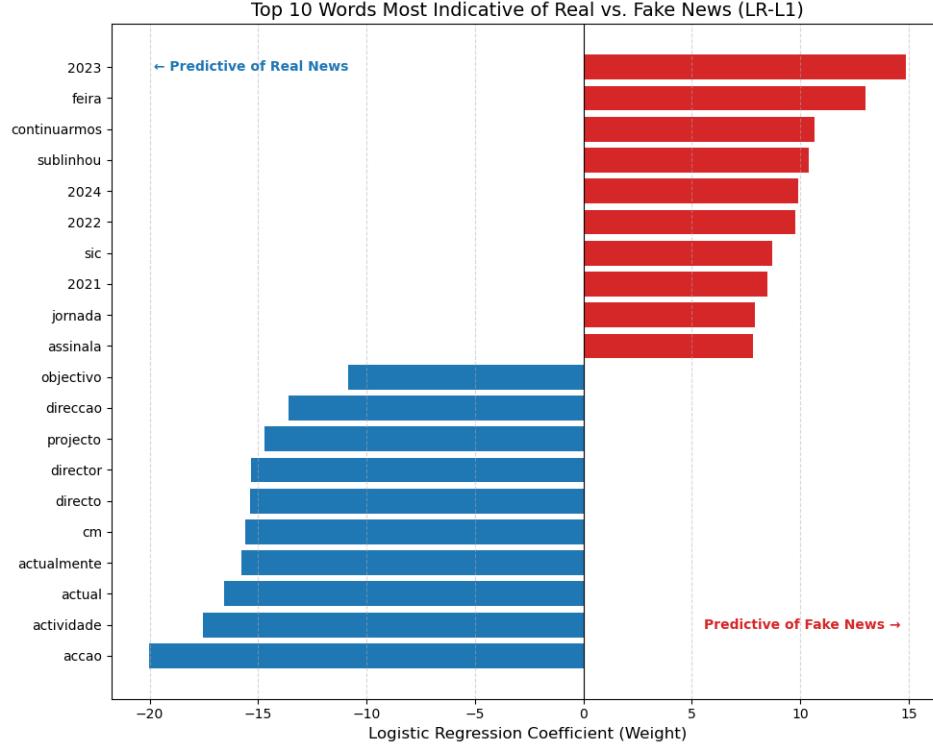


Figure 2: Top 10 words most indicative of Real vs. Fake news, extracted from the Logistic Regression (L1) model.

## 6.2 Regularization Comparison: L1 vs. L2

A quantitative comparison was conducted to analyze the impact of regularization type on model complexity and feature selection. Table 3 details the number of non-zero weights retained by each model type from the initial feature space of 5,000 terms.

Table 3: Comparison of non-zero features retained by L1 vs L2 regularization.

Model	Total Features	Non-Zero Weights	Features Removed
Logistic Regression (L1)	5000	2102	2898
Logistic Regression (L2)	5000	5000	0

The L1 (Lasso) regularization induced significant sparsity, driving 2,898 features (approximately 58% of the vocabulary) to exactly zero. This effectively performs automatic feature selection, eliminating irrelevant or redundant terms while maintaining high predictive performance. In contrast, L2 (Ridge) regularization retained all 5,000 features, shrinking their coefficients towards zero without eliminating them entirely.

This distinction dictates the preference for each method in text classification tasks. L1 regularization is preferred when interpretability and model compactness are priorities, as it produces a sparse model that identifies the most critical words. It is also beneficial when the feature space is extremely

high-dimensional and expected to contain significant noise. Conversely, L2 regularization is more suitable when highly correlated features are present or when it is believed that small contributions from all features are necessary to capture the full semantic context, preventing the loss of potentially useful information.

### 6.3 Local Interpretability with LIME

To examine model behavior at the instance level, Local Interpretable Model-agnostic Explanations (LIME) were generated for four specific samples (IDs 2921, 2437, 5557, 1697). Figure 3 illustrates the local feature contributions for both the Logistic Regression (L1) and MLP models.

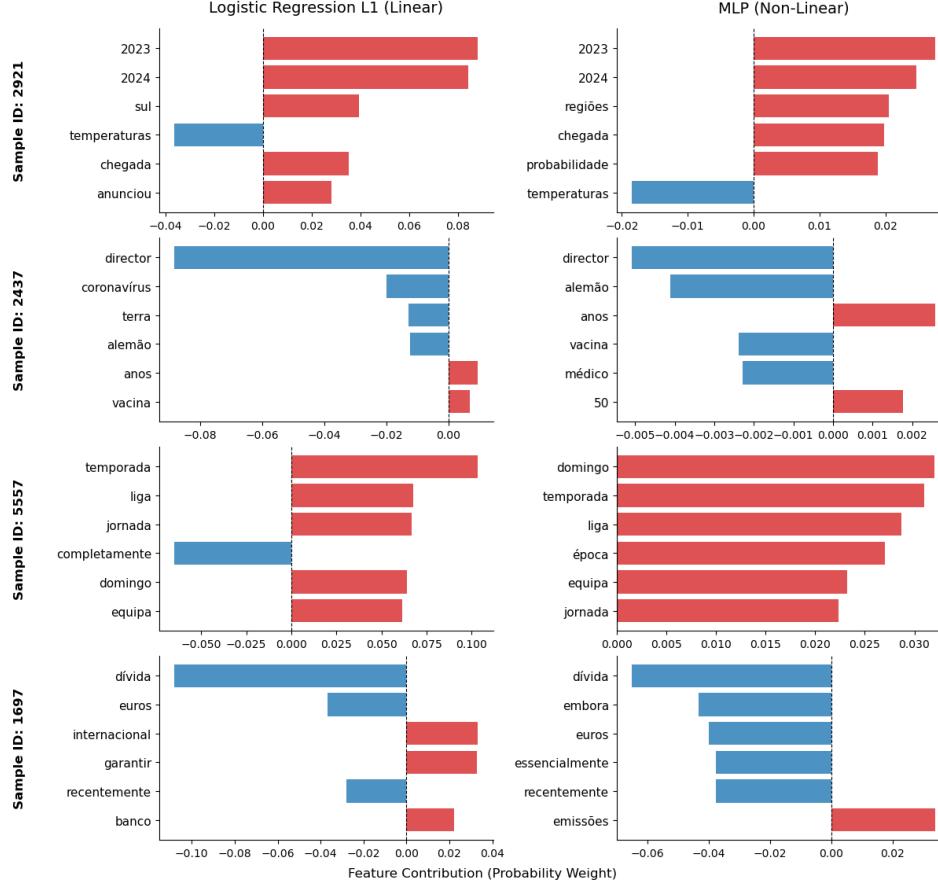


Figure 3: Side-by-side LIME explanations for Logistic Regression (L1) and MLP on four validation samples. The bars represent the contribution of specific words to the local prediction.

The visualization highlights the contrast between the linear decision boundaries of the Logistic Regression model and the non-linear complexity of the MLP. While the linear model relies on a fixed set of global coefficients, the MLP’s local explanations vary more dynamically, weighing terms differently depending on the specific context of the article. This allows the MLP to capture more nuanced semantic relationships, though it can occasionally lead to less intuitive feature attributions compared to the linear baseline.

### 6.4 Global vs. Local Analysis: Permutation Importance

To complement the local insights provided by LIME, global feature importance for the MLP model was assessed using Permutation Importance. This method involves randomly shuffling the values of a single feature across 1,000 random validation samples and measuring the resulting decrease in model performance. A large drop in performance indicates that the model relies heavily on that feature for accurate classification.

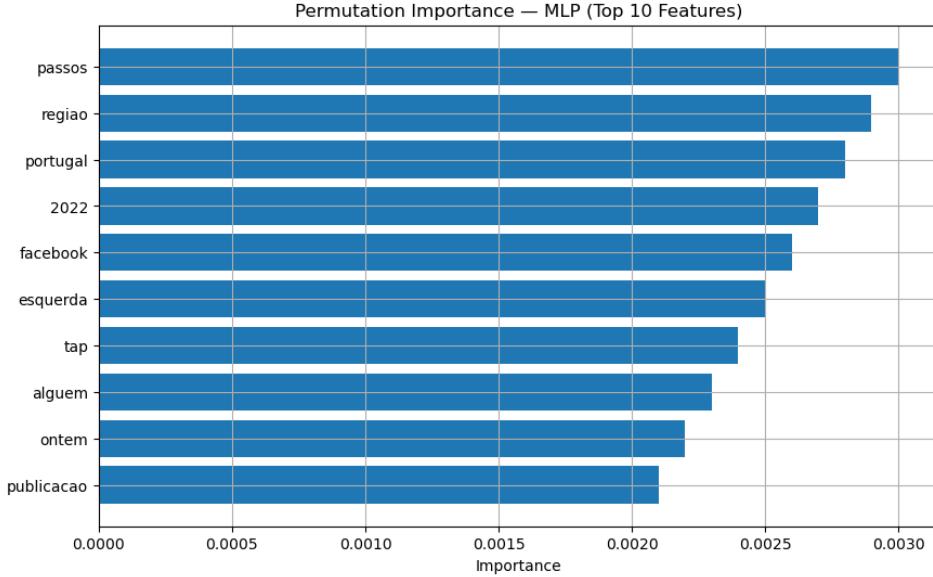


Figure 4: Top 10 features by Permutation Importance for the MLP model, calculated on 1,000 random validation samples.

These methods offer complementary insights: LIME provides *local* explanations by identifying features driving specific predictions, whereas permutation importance reveals *global*, dataset-wide dependencies. While discrepancies naturally exist—globally important features may not drive individual instances—combining these approaches ensures a holistic understanding of model behavior.

## 7 Unsupervised Analysis: Clustering

To explore latent semantic structures independent of labels, we applied K-Means clustering, which partitions data by minimizing the within-cluster sum of squares (WCSS). The process used the TF-IDF training representations defined in Section 3 (5,000 features) with the number of clusters fixed at K=5, per challenge requirements.

### 7.1 Cluster Interpretation

Following the assignment of documents to clusters, the three documents closest to each centroid were inspected to infer the semantic meaning of each group. Table 4 maps the assigned semantic labels to the representative articles and key terms found in each cluster.

Table 4: Semantic inspection of K-Means clusters based on top keywords and representative documents.

Label & Key Terms	Representative Snippets (Row IDs)
<b>Geopolitics (Ukraine War)</b> ucrania, russia, russo, guerra, putin, wagner, moscovo, russa, kiev, nato	<b>Row 22308:</b> Líder do Grupo Wagner quer batalha final para clarificar situação na Ucrânia: “Em teoria, a Rússia já pôs um fim a isto”... <b>Row 15728:</b> Compreender o conflito: o que podemos esperar da contraofensiva ucraniana? Como será a contraofensiva ucraniana?... <b>Row 43637:</b> Dia da Rússia. Putin visitou soldados feridos e entregou condecorações...

*Continued on next page...*

*Continuation of Table 4*

Label & Key Terms	Representative Snippets (Row IDs)
<b>Public Health &amp; Society</b> <i>pessoas, saude, portugal, anos, dia, ainda, pode, feira, ter, vai</i>	<b>Row 35841:</b> Manuel Carmo Gomes. "Não nos vamos ver livres da covid por várias razões". "O mais provável é que venhamos a recomendar a vacinação..." <b>Row 3938:</b> Estado não vai ter dinheiro para pagar reformas daqui a 15 anos. O alerta é lançado por um economista... <b>Row 44810:</b> "Estado não vai ter dinheiro para pagar reformas daqui a 15 anos". Para o economista vai haver um grave problema social...
<b>Domestic Politics</b> <i>governo, ministro, costa, ps, antonio, psd, presidente, partido, primeiro</i>	<b>Row 31135:</b> Ana Gomes: "Este governo também está a fabricar populismos". Militante de base, Ana Gomes é uma das vozes independentes... <b>Row 39849:</b> Maioria dos conselheiros defende remodelação. O Presidente Marcelo anunciou a convocação do Conselho de Estado... <b>Row 154:</b> "Para não ter esta governação socialista, têm de votar PSD". Vice-presidente social-democrata explica projeto do partido...
<b>Economy &amp; Finance</b> <i>euros, milhoes, mil, ano, portugal, valor, banco, divida, aumento</i>	<b>Row 6924:</b> Governo gasta dois mil milhões, mas tem margem para ir até 3,5 mil milhões. IVA a zero em alimentos essenciais... <b>Row 40474:</b> Infraestruturas de Portugal e Metropolitano de Lisboa são os maiores beneficiários do PRR. No total do Plano contabilizam-se... <b>Row 7143:</b> "[Há] 880 milhões de euros que fogem para offshores."...
<b>Curiosities &amp; Clickbait</b> <i>anos, mae, jovem, homem, mulher, vida, policia, crianca, pai, filho</i>	<b>Row 2716:</b> 5 casos curiosos de crianças que afirmam se lembrar de suas vidas passadas. O que acontece quando morremos?... <b>Row 37492:</b> 5 casos curiosos de crianças que afirmam se lembrar de suas vidas passadas. O que acontece quando morremos?... <b>Row 1828:</b> 5 casos curiosos de crianças que afirmam se lembrar de suas vidas passadas. O que acontece quando morremos?...

## 7.2 Visualization via PCA

Figure 5 displays a 2D PCA projection of the 5,000-term feature space. The left plot confirms that K-Means ( $K = 5$ ) identifies distinct semantic regions. The right plot, colored by ground truth, reveals that while "Fake" news (blue) shows higher density in the bottom-left quadrant, it significantly overlaps with "Real" news (red). This indicates that misinformation is not semantically isolated but is dispersed across diverse topics.

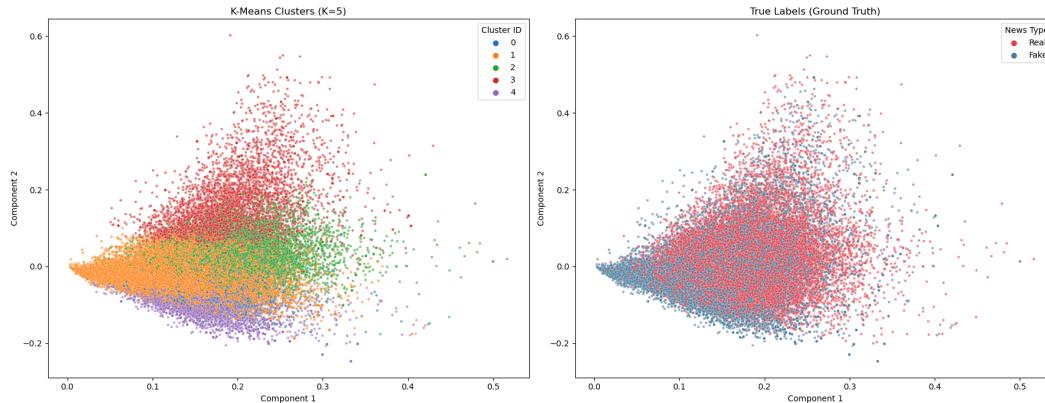


Figure 5: 2D PCA projection. Left: K-Means clusters. Right: True Labels (Red=Real, Blue=Fake).