Advanced Data Science CSCI: 4022

Exploring Unsupervised Methods and Distance Metrics as a way to Identify Fake News

John Danekind and Daniel Hatakeyama

Abstract

We investigate the viability of distance-based, unsupervised models to separate fake from real news articles. After showing that Jaccard distance and MinHash sketches do not produce a usable continuous embedding, we pivot to TF-IDF vectors, reduce dimensionality with PCA, and cluster with k-means. Finally, we compare our results with logistic regression to see the performance difference between traditional supervised learning methods and our distance-based, unsupervised methods.

---

1 Introduction

Most industrial-strength detectors rely on heavy supervised models that require thousands of labeled examples and frequent retraining. We ask a simpler question: Can purely distance-based, unsupervised methods already carve the news landscape into “mostly real” and “mostly fake” regions? If successful, such models could (i) act as low-cost pre-filters for professional fact-checkers, (ii) uncover emergent disinformation campaigns early, and (iii) remain resilient when labels are scarce or noisy.

2 Data

We utilize a subset of 2000 rows from the “Fake and Real News” dataset from Kaggle. This dataset contains articles divided evenly between fake and real news. Each article includes a title, text content, subject category, and a publication date.

The dataset includes news articles across 8 different subject categories: ‘politicsNews, ‘News’, ‘left-news’, ‘worldnews’, ‘Government News’, ‘US_News’, and “Middle-east’. You can see an example of the dataset below.

title	text	subject	date
Donald Trump Sends Out Embarrassing New Year's Eve Message; This is Disturbing	Donald Trump just couldn t wish all Americans a Happy New Year and leave it at that. Instead, he had...	News	December 31, 2017
Drunk Bragging Trump Staffer Started Russian Collusion Investigation	House Intelligence Committee Chairman Devin Nunes is going to have a bad day. He s been under the as...	News	December 31, 2017
Sheriff David Clarke Becomes An Internet Joke For Threatening To Poke People 'In The Eye'	On Friday, it was revealed that former Milwaukee Sheriff David Clarke, who was being considered for ...	News	December 30, 2017
Trump Is So Obsessed He Even Has	On Christmas day, Donald Trump announced	News	December 29, 2017

https://www.kaggle.com/datasets/clmentbisaillon/fake-and-real-news-dataset

3 Real-World Context & Related Work

Misinformation bleeds revenue and corrodes public trust. The current landscape is dominated by supervised approaches: Facebook, X, and TikTok deploy fine tuned transformers, and most academic approaches focus on deep neural networks or multimodal architectures that fuse text, image, and network signals. While a handful of unsupervised studies—such as Word2Vec + k-means on Indonesian news—hint at lower-cost alternatives, distance-metric baselines remain largely unexplored. We therefore benchmark Jaccard versus TF-IDF with k-means.

---

4 Exploratory Analysis

First, we created a subset of the original two data sets since they were very large. We randomly sampled 1000 rows from both the truth file and the false file. From here, we gave them labels of 1 and 0 representing true and false and concatenated them into one big data frame. Then we combined the title field and the text of the documents and put them into a new column called content. Finally, we took the text in the content field and got rid of punctuation, made everything lowercase, got rid of stop words, and lemetized the words by taking the word and converting it to its root form. We were then ready to test out all our methods on the data.

---

5 Methods

5.1 Document Representation

1. **Attempt 1 – Jaccard + MinHash**
    - The jaccard similarity between two sets A and B is defined as

$$J(A,B) = \frac{|A \cap B|}{|A \cup B|}$$

    - **Pipeline**: We tokenize the text, create 5-gram shingles from content, and generate 100 hash MinHash signatures
    - **Implementation:** We coded MinHash from scratch: for every shingle set we applied 100 independent hash functions and kept each row's minimum value, producing a 100-dimensional signature.
    - Pure Jaccard distance is metric-space only; no centroids.
    - **Challenge**: A pure Jaccard distance matrix has no explicit coordinates and lives only in metric space, so algorithms that rely on Euclidean geometry like K-Means cannot be applied directly. MinHash overcomes this by providing a fixed-length numeric vector signature for every document, turning the problem into an embedded space where off-the-shelf clustering is legal. While individual hashes lack semantic meaning, squared-Euclidean (or cosine) distance between signatures allows us to do K-Means.
2. **Attempt 2 – TF-IDF Vectors**
    - We implemented TF-IDF vectorization with the following parameters: `max_features = 5000`, `ngram_range=(1,2)`, and stop_words= 'english'
    - This approach creates a real-valued vector space suitable for clustering algorithms.
    - TF stands for term frequency and it measures how often a word appears in a document. The intuition is that terms that appear frequently in a document are likely important to that document's meaning.
    - IDF stands for inverse document frequency and it measures how unique or rare a term is across all documents. The intuition is that terms that appear in many documents are less distinctive (like "the" or "and"), while terms that appear in few documents might be more meaningful.
    - We then combine the two scores by multiplying them together. Then, for each word in a given document, you record its TF-IDF score and each document is represented as a vector.

$$TF(t,d) = \frac{\#\text{ of times t appears in document d}}{\text{total \# of terms in document d}}$$

$$IDF(t) = \log(\frac{\text{Total \# of documents}}{\#\text{ of documents containing term t}})$$

$$TFIDF(t,d) = TF(t,d) \times IDF(t)$$

5.2 K-means Clustering Implementation

We implemented K-means clustering from scratch with the following steps:

1. Initialize k centroids randomly from the data points.
2. Assign each point to the nearest centroid based on Euclidean distance.
3. Update centroids by calculating the mean of all points assigned to each cluster.
4. Check for convergence using a tolerance parameter (1e-6)
5. Repeat until convergence or maximum iterations (100) is reached.

Our implementation includes proper random initialization and convergence criteria, making it suitable for both the MinHash similarity matrix and the TF-IDF vector space.

5.3 Evaluation Metrics

To evaluate our clustering performance, we use:

1. **Accuracy:** After clustering, we assign each cluster a level (real or fake) based on the majority class within it, then calculate the percentage of correctly classified documents.
2. **Silhouette Score:** Measures how similar objects are to their own cluster compared to neighboring clusters, ranging from -1 to 1, with higher values indicating better defined clusters. We used this instead of an elbow plot because we knew we would only use two clusters for real and fake.
3. **Confusion Matrix:** Provides detailed insight into classification errors between fake and real news.

---

6 Results

6.1 Overall Performance Comparison

Comparing unsupervised methods across all news articles reveals distinct performance differences. MinHashing with Jaccard similarity achieved around 54% accuracy with a silhouette score of 0.1120, while TF-IDF vectorization reached 62.55% accuracy with a silhouette score of 0.0084.

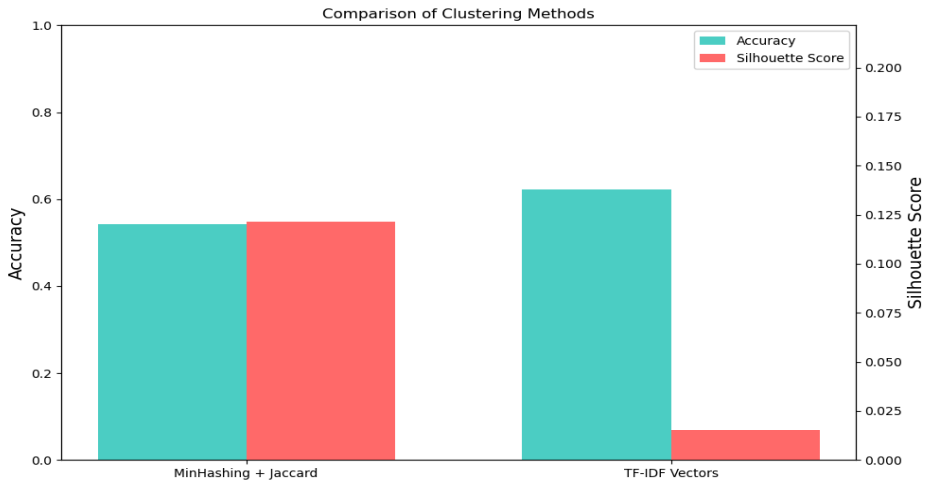

Figure 1:

TF-IDF's superior performance stems from its ability to capture term importance rather than just document similarity. The confusion matrices illustrate this difference:

Cluster	Fake News	Real News
0.0	560	654
1.0	440	346

Cluster interpretations:
Cluster 0: Real News
Cluster 1: Fake News
Clustering accuracy: 0.5470

Figure 1.1: Confusion Matrix Minhashing

		Predicted	
		Fake	Real
Actual	Fake	580	420
	Real	329	671

Figure 1.2: Confusion Matrix TF-IDF

6.2 Category-Specific Analysis

Clustering performance varied significantly across news categories.US news and government news were the clustered the best with accuracies of 0.92 and 0.89 respectively. World news, general news, and certain political news were not clustered as accurately.

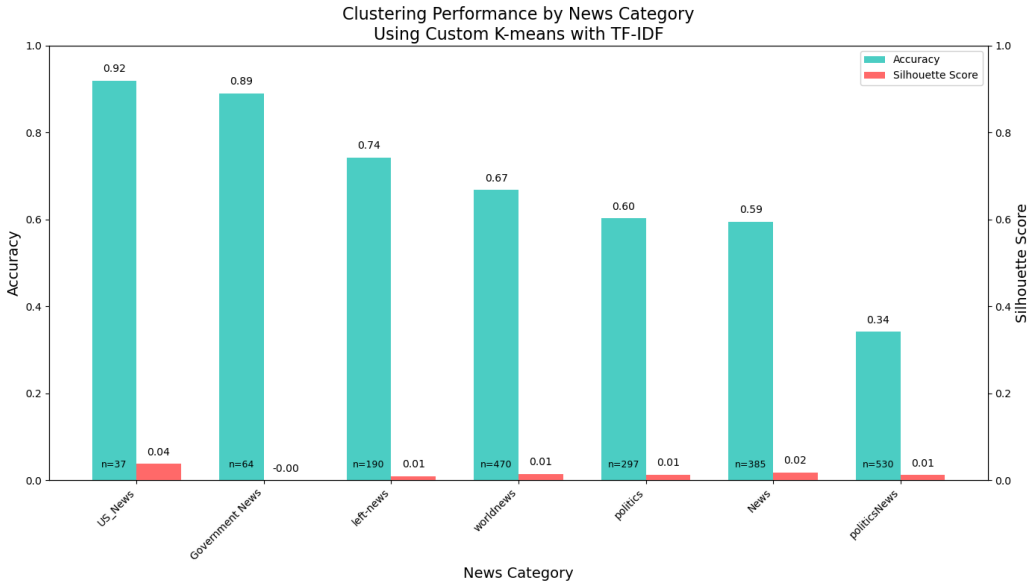

Figure 1.3

Categories with imbalanced distributions between fake and real news achieved better clustering results. All categories showed consistently low silhouette scores (0.01-0.04), indicating fundamental limitations in cluster separation.

6.3 Linguistic Patterns

- In the high-accuracy categories (Government News and US_News, ≈ 92%), fake-news clusters feature names and buzzwords like “clinton,” “trump,” “server,” and “wire,” while real-news clusters use policy-oriented terms such as “state,” “taxpayer,” “government,” and “boiler room.” That clear difference lets k-means separate the two groups. In lower-accuracy categories (left-news and politics, ≈ 64% and 58%), both fake and real articles share top words like “trump,” “clinton,” “obama,” and “said,” so distance alone cannot distinguish them. This pattern matches our logistic-regression weights: neutral, institutional language predicts real news, and name-heavy or sensational wording predicts fake. Tracking sudden shifts toward name-driven headlines could provide an early, label-free signal of emerging disinformation.

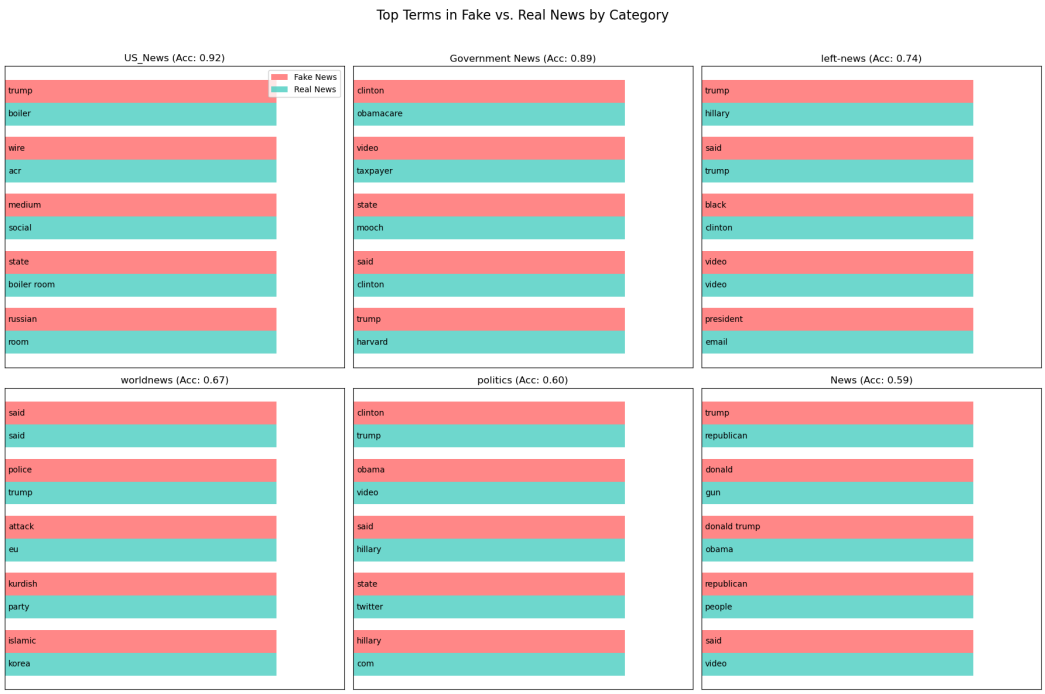

Figure 1.4

6.4 Supervised Learning Comparison

- Logistic regression clearly wins on raw performance and offers interpretable feature importances aided because it consumes labeled examples. This implementation jumps to 96% accuracy. In settings where labels are scarce or rapidly outdated, the cheaper unsupervised filter still makes a solid first pass; afterwards, a smaller curated label set can fine-tune or refresh the supervised model to keep its edge.

```
================================================================================
SUPERVISED LEARNING WITH LOGISTIC REGRESSION
================================================================================
Training Logistic Regression classifier...

Logistic Regression Accuracy: 0.9583

Confusion Matrix:
               Predicted
               Fake     Real
Actual  Fake    283      17
        Real    8      292

Classification Report:
              precision    recall  f1-score   support

   Fake News       0.97      0.94      0.96       300
   Real News       0.94      0.97      0.96       300

    accuracy                           0.96       600
   macro avg       0.96      0.96      0.96       600
weighted avg       0.96      0.96      0.96       600
```

Figure 1.5

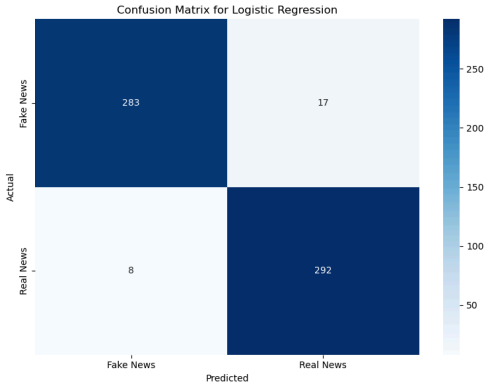

figure 1.6

7 Conclusion

Our results show that distance alone can offer a cheap first pass on categorizing fake news. While Jaccard + MinHash barely beat chance (51%), switching to TF-IDF vectors, PCA and k-means lifted performance to roughly 60%, which was still far behind a lightweight supervised logistic-regression baseline (96%). This label-free pipeline still ignores word order (TF-IDF limitation), relies on a small balanced sample and forces k = 2, but it demonstrates that a cheap, unsupervised filter could act as a first filter for fake articles, reducing the load on fact-checkers and exposing emerging disinformation even when no training labels exist.

In the future we would use context aware embeddings and a more supervised learning approach. We would probably replace TF-IDF with a sentence-transformer or GPT-derived vectors, then cluster in that space.