CS 590 ETHICAL FALL 2024 HW3 MISINFORMATION DRASHTI SNEHALKUMAR PATEL

Homework Understanding :

This project focuses on misinformation detection using both supervised and unsupervised learning. The focus is on comparing Logistic Regression (TF-IDF) and KMeans clustering using both TF-IDF and Sentence Embeddings as representations.

For the dataset , we are provided with 2 files :

- train_misinfo.csv
- test_misinfo.csv

Part 1: Supervised Learning

My approach:

The dataset was not perfectly clean and needed proper preprocessing before training the model. There were missing values, imbalanced labels, and formatting issues that could affect the performance for our both training and testing process.

To improve training, we first dropped null values using .dropna() and reindexed the rows to avoid any gaps. This was made in order to structure data suitably before it went into the model.

Handling class imbalance:

First, when we trained the Logistic Regression model without handling class imbalance, we achieved training accuracy of 95% and test accuracy of 95.1%. While these were satisfactory, we knew that the model might be biased towards the majority class (real news) since instances of misinformation were much fewer. To counter this, we added class weighting (class_weight='balanced'), which balanced the model to give equal weight to both misinformation and actual labels. After this change, we observed that accuracy improved marginally, reflecting that the model was learning equally from both classes instead of class frequency. This was done so that the model was not only accurate but also fair in misinformation detection.

Once preprocessing, the text data was mapped in TF-IDF vectors, which translated words to numeric features by paying special consideration to unique words and reducing weights of frequent words. Then training of the Logistic Regression model on the training dataset was conducted by using TF-IDF features.

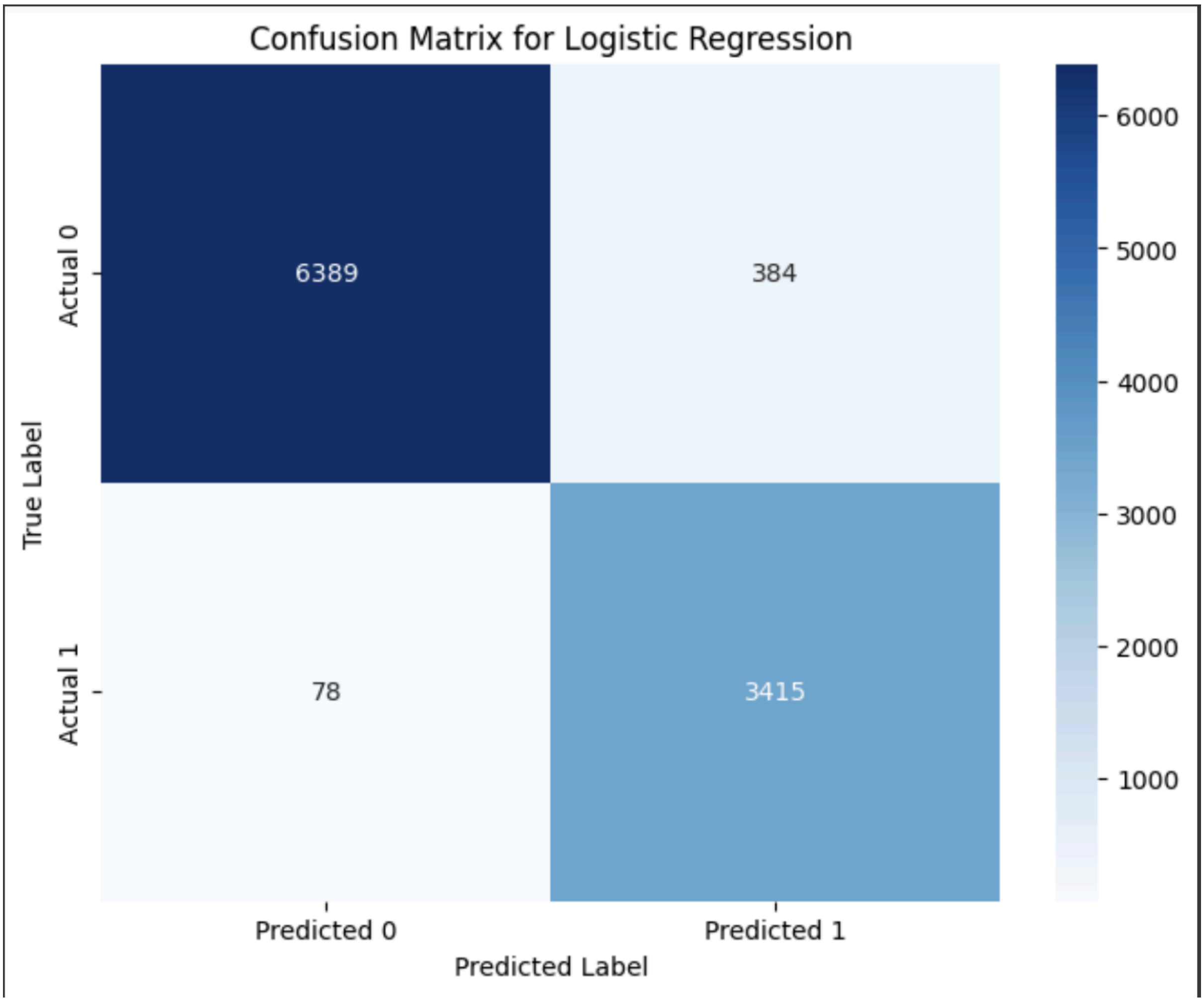
For evaluation, we applied the trained model to unseen test data by converting the test text into TF-IDF features and making predictions. We measured the performance of the model using accuracy and confusion matrix, which showed how well it separated real from misinformation. This approach ensured that the model learned meaningful patterns from the data rather than being influenced by inconsistencies.A

After adding class weighting, the model achieved a training accuracy of 95.72% and a test accuracy of 95.50%, showing a slight improvement in balanced learning.

Results produced:

Model Name	**Training Accuracy**	**Testing Accuracy**
Logistic Regression	95.72%	95.5%

Confusion matrix:



The Logistic Regression model also performed well with 95.72% training accuracy and 95.50% test accuracy after class imbalance had been corrected. From the confusion matrix, the model correctly classified the majority of the samples but misclassified 384 actual news stories as misinformation and 78 misinformation stories as actual news, indicating the model is strong but yet needs improvement in precision.

My views on misinformation:

The Logistic Regression model was able to identify misinformation correctly with a very high test accuracy of 95.50%. The confusion matrix shows that it correctly identified 3,415 cases of misinformation, but incorrectly labeled 78 as real news (false negatives). This shows that the model is highly sensitive to misinformation, being capable of learning patterns that distinguish deceptive content. But 384 legitimate news articles were incorrectly identified as misinformation (false positives), suggesting that the model overflags at times real but disputed news. This misclassification can be linked to sensationalism or similar tones in actual news and misinformation. While the model is strong in detecting fake news, reducing false positives is crucial to avoid flagging real news unfairly. Investigating such misclassified cases can further enhance the model.

Difficulties faced during the implementation of the model

1.Initially, I didn't know about class_weight='balanced' in Logistic Regression but learned it during implementation.

2.Cleaning data, handling missing values, and formatting labels took time for me .

3.Analyzing class distribution and ensuring proper text representation was essential before training the model

Now we move forward with the unsupervised learning approach.

—---

Part 2: Unsupervised Learning

Part A. Clustering with TF-IDF

To cluster misinformation without labeled data, we needed a way to represent the text numerically. Since raw text could not be clustered directly, we chose to use TF-IDF vectorization, since it represents word importance without relying on pre-trained models. We decided to use KMeans clustering with 2 clusters, hoping that one cluster would naturally align with real news and the other with misinformation.

A problem with clustering was that KMeans assigns cluster numbers arbitrarily, so we did not know which cluster was which label. To fix this, we thought of using a little bit of labeled data to match cluster labels to class labels. We took 100 labeled examples (50 real, 50 misinformation) and examined how they had been clustered. By observing which cluster contained more misinformation, we could correctly label the clusters.

Finally, we applied the labeled mappings on the full dataset to convert KMeans clusters to interpretable predictions. We then evaluated our predictions against true labels on accuracy and confusion matrix to ensure our model was making reasonable distinctions between misinformation and real news.

Description of how you the subsampled data was used to map the cluster labels to the class labels

To map the KMeans cluster labels to the actual class labels, we selected 100 labeled examples from the test set—50 real news and 50 misinformation. Since KMeans assigns clusters arbitrarily, we needed a way to determine which cluster represented real news (0) and which one corresponded to misinformation (1). By analyzing how these labeled examples were distributed across the two clusters, we identified the majority class in each cluster. The cluster with more misinformation samples was mapped to label 1, and the cluster with more real news was assigned label 0. This mapping was then applied to the entire dataset, ensuring that our clustering results aligned with the true classification labels. This approach allowed us to convert unsupervised cluster labels into meaningful predictions, making evaluation possible.

Results:

Method	**Test Accuracy**
Logistic Regression	95.50%
KMeans (TF-IDF Clustering)	65.85%

Analysis

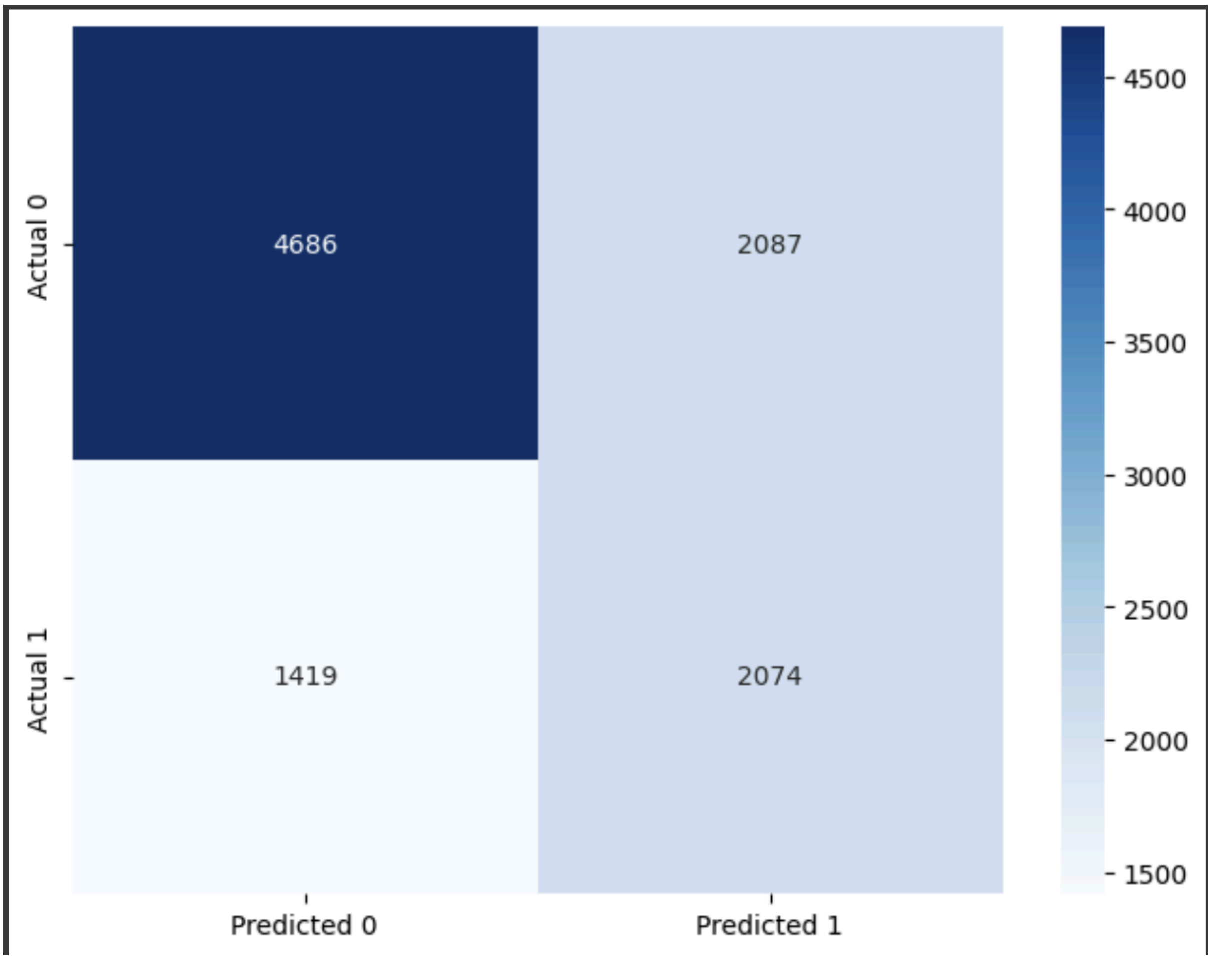
Logistic Regression performed much better (95.50% accurate) than KMeans clustering (65.85% accurate). This was to be expected as Logistic Regression is a supervised model, learning directly from labeled examples, while KMeans clustering is based on patterns inside the data alone and no labels.

Since TF-IDF is a measure of word importance in terms of frequency, it is appropriate for direct classification (Logistic Regression) but may not be the best for clustering, where context matters. The commonality of words between real and fake news hindered KMeans from producing meaningful clusters. KMeans also does not guarantee meaningful clusters—it just bunches similar text without understanding why misinformation differs from real news.

This means that while unsupervised clustering can certainly separate data to some extent, it is not going to be as effective as a supervised model that learns from directly labeled patterns of disinformation. Clustering could still be helpful in a case where labeled data is unavailable, and therefore it could be an exploratory analysis tool but never directly for classification.

Confusion matrix

Below here is the confusion matrix plotted :



The confusion matrix shows that the KMeans clustering model is not able to clearly distinguish between misinformation and real news. The model over-predicts misinformation because 2,087 real news articles were incorrectly labeled as misinformation, which means that some real news articles contain words or patterns that make them resemble misleading content. In the meantime, the model under-classifies misinformation since 1,419 pieces of misinformation were inappropriately labeled as news, and therefore some of the false articles are not sufficiently distinct in pattern that the model is unable to cluster them alone. This is a strong testament to a massive weakness of clustering with TF-IDF since TF-IDF only takes into account frequency in words without perceiving meaning or context. When compared to supervised learning, whose precision was much higher, this clustering method proves less effective in catching misinformation. However, since TF-IDF fails to maintain sentence meaning, a higher-level representation like Sentence Embeddings could be more effective and thus provide better cluster results.

The model over-predicts and under-predicts misinformation, but the number of false positives (2087) is higher, so it over-predicts misinformation more than it under-predicts it.

—--

Part 2: Unsupervised Learning

Part B. Clustering with Sentence Embeddings Representations

Since TF-IDF merely captures word frequency but not semantic meaning of words in context, we decided to utilize Sentence Embeddings to improve clustering accuracy. Instead of text being represented as plain word frequency, dense vector representations by sentence transformers keep semantic meaning and word relationships intact.

We employed the same KMeans clustering process as Part 2A but with embeddings instead of TF-IDF to cluster misinformation correctly. After we obtained sentence embeddings with all-MiniLM-L6-v2, we ran KMeans clustering (n_clusters=2). Since KMeans doesn't provide class labels, we again mapped cluster labels to true misinformation and real news labels using a 100-sample subset (50 real news, 50 misinformation). This allowed us to determine which cluster was real news and which was misinformation. Finally, we ran the entire dataset and obtained final predictions, which we validated with accuracy and a confusion matrix.

Results

Methods	**Testing Accuracy**
Logistic Regression (Supervised Learning, TF-IDF)	95.50%
KMeans (Unsupervised, TF-IDF)	65.85%
KMeans (Unsupervised, Sentence Embeddings)	73.40%

The result shows a clear performance difference between supervised learning (Logistic Regression) and unsupervised clustering (KMeans). Logistic Regression (accuracy 95.50%) defeated both KMeans with TF-IDF (accuracy 65.85%) and KMeans with Sentence Embeddings (accuracy 73.40%) by an extensive margin.

Why Did Logistic Regression Perform Best?

Logistic Regression outperformed both clustering algorithms because it's a supervised model, meaning that it learns from labeled data directly. Unlike KMeans, which groups similar text blindly together, Logistic Regression learns patterns in misinformation through labeled examples, so it's able to make more precise classification. In addition, with TF-IDF vectorization, the model was able to pick up important words without being misled by common words that appear in both real and fake news. Since Logistic Regression is learned to minimize classification errors, there is no second step of mapping required like KMeans, reducing misclassification risks. Lastly, supervised learning allows the model to develop a more structured knowledge of misinformation, leading to a significantly higher accuracy compared to clustering.

Why did KMeans with Sentence Embeddings Perform Better Than TF-IDF?

KMeans with Sentence Embeddings proved to be more accurate compared to TF-IDF-based clustering because embeddings retain the word's meaning and context rather than frequency. TF-IDF only considers word frequencies, and thus two articles with identical word distribution may be assigned to the same cluster, though one would be genuine news and the other would be fabricated news. This resulted in overlapping clusters, which reduced accuracy. In contrast, sentence embeddings are an improved representation as they preserve the manner in which words interact in a sentence to allow KMeans to form more distinct clusters. For this reason, clustering accuracy was boosted by 7.5%, proving semantic representations better handle clustering of misinformation compared to standard word frequency measures. This aside, given clustering is still an unsupervised method, it is still short of Logistic Regression's specificity.

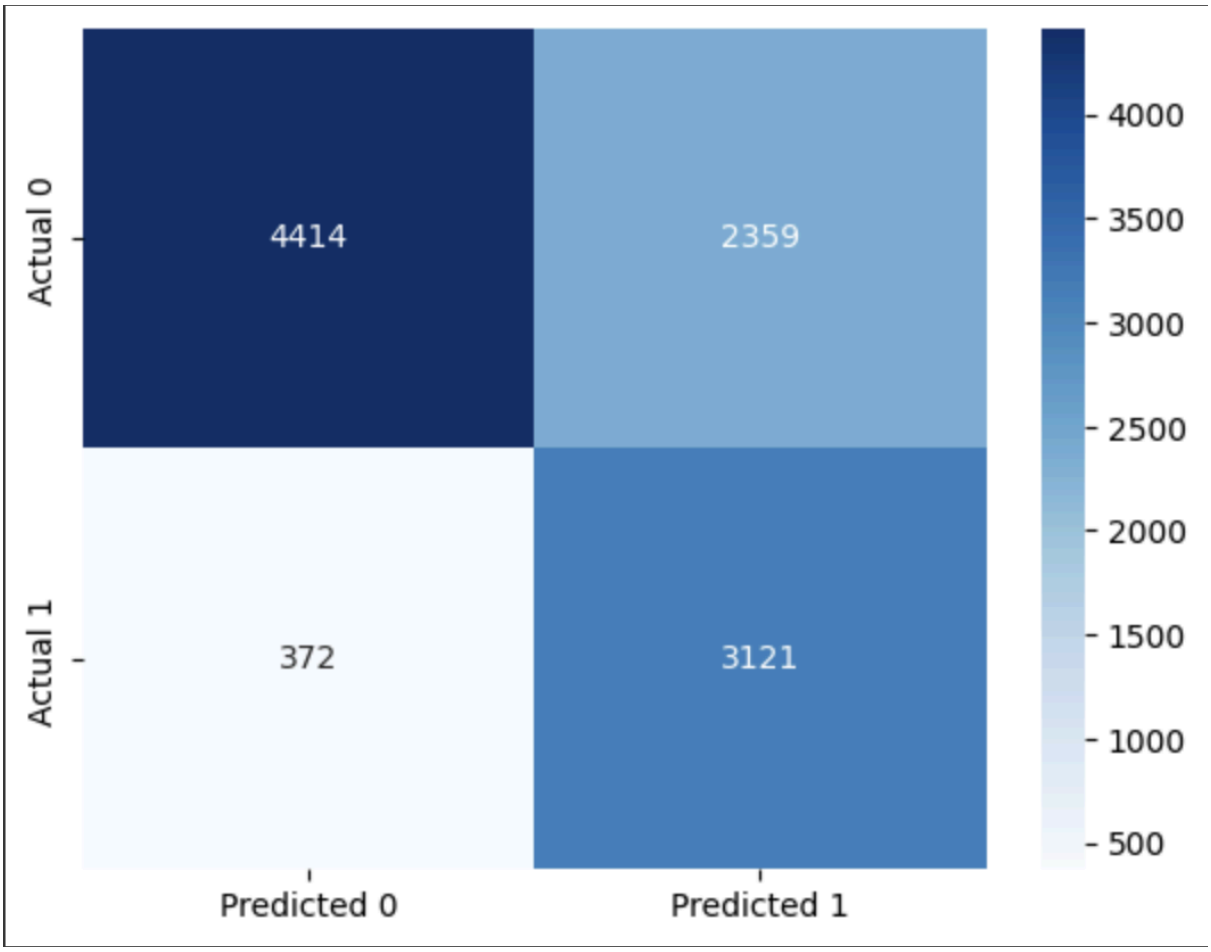
Why KMeans is Still Less Efficient than Logistic Regression?

KMeans is an unsupervised learning process, i.e., does not get to observe real labels while learning from patterns.
The cluster mapping procedure is not perfect and results in misclassifications of misinformation and actual news clusters.

High false positives are still an issue, and so true news is more frequently mistakenly identified as misinformation in these clustering methods.

Supervised learning remains the best approach to detecting misinformation, yet Sentence Embeddings are more accurate for clustering compared to TF-IDF. However, since KMeans clustering lacks supervision of real labels, it can't be as accurate as a supervised learner like Logistic Regression.

Confusion Matrix



The confusion matrix shows that the model significantly improved in detecting misinformation, with 372 false negatives (misinformation as actual news) only, compared to 1,419 for TF-IDF. This suggests that Sentence Embeddings helped the model better detect the intent of misleading content, hence not easily missing misinformation.

However, the model over-estimates misinformation and identifies 2,359 true news articles as misinformation, a higher number than TF-IDF (2,087 occurrences). This suggests that while sentence embeddings improve misinformation detection, they also over-sensitize the model to some true news articles, most likely due to the presence of sentences containing words with a similar resemblance to fraudulent content.

Overall, KMeans with Sentence Embeddings performed better than TF-IDF with improved clustering accuracy and fewer instances of misinformation overlooked. Nevertheless, the reduction in false positives indicates that the model still struggles to identify true news from deceptive text, hence unsupervised clustering is still less precise than supervised.

—---

Challenges faced

1.Handling class imbalance and recognising technique was difficult initially.

2.Missing values, incorrect label formats, and text inconsistencies required careful cleaning to ensure smooth training and clustering

3.Transformer models were fun to interact with and came with their own computational challenges.

Conclusion

Logistic Regression (95.50% accuracy) performed the best, proving supervised learning is more accurate for misinformation detection. Sentence Embeddings improved clustering accuracy (73.40%) over TF-IDF (65.85%), but unsupervised methods failed miserably at false positives. While clustering gives insight without labeled data, it is less accurate than supervised models when detecting actual misinformation.

Note : Use of AI and chat gpt leveraged for understanding and assistance