NLP Challenge

OVERVIEW

Objective: Build a model to classify news as *real (1)* or *fake (0)* using NLP.

Scope:

- Data preprocessing and text cleaning.
- Model experimentation with multiple classifiers.
- Selection of best-performing model for final predictions.

Key Milestones:

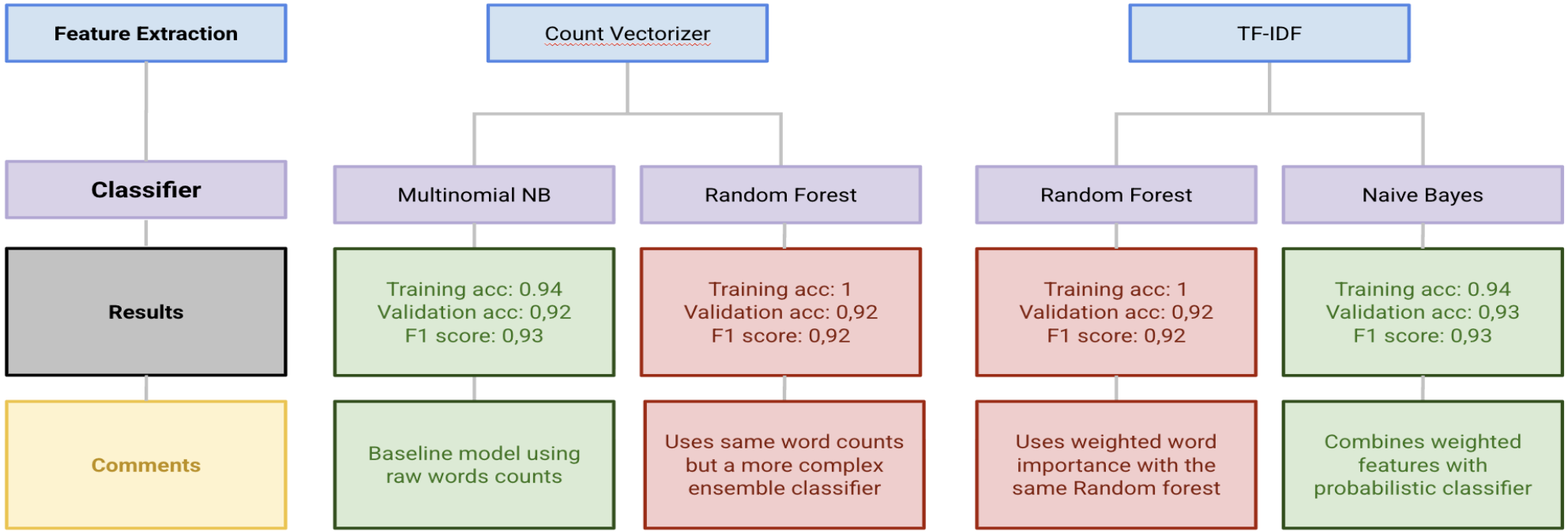
- Implemented full preprocessing pipeline (tokenization, stopword removal, lemmatization, stemming).
- Compared Count Vectorizer and TF-IDF approaches.
- Evaluated models on training and validation datasets.

EXECUTIVE SUMMARY

- **Final Model:**
  - TF-IDF + NAIVE BAYES
- **Training accuracy:** 0.94%
- **Validation accuracy:** 0.93%
- **Other models:**
  - Count Vector + Multinomial (base model)
  - Count Vector + Random Forest
  - TF-IDF + Random Forest

METHODS OF PREPROCESSING

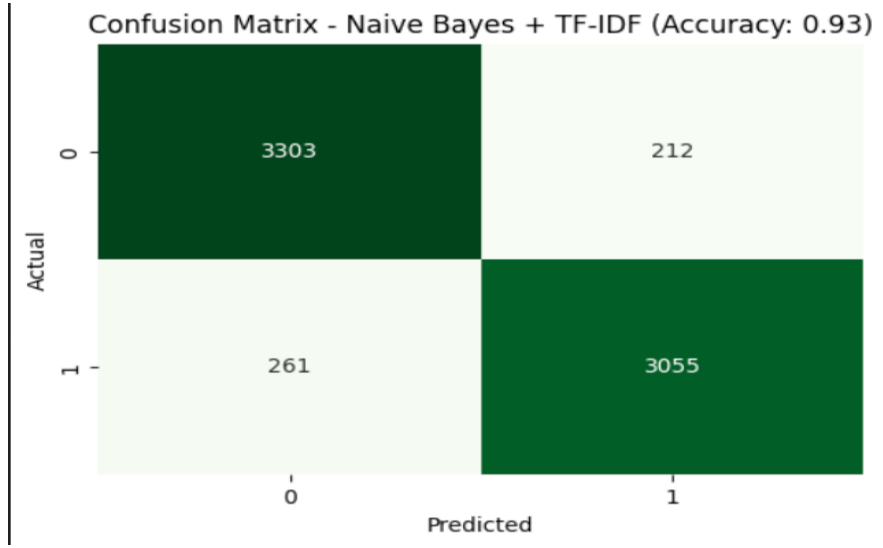
- We used a function that tokenizes, removes stopwords, applies lemmatization and stemming, and cleans text. We apply it to the entire dataset.
- The preprocessing function prepares raw text data for machine learning by performing several key steps:
  - **Tokenization:** Splits the text into individual words (tokens) so each word can be analyzed separately.
  - **Stopword Removal:** Removes common words like “the”, “is”, “and” that don’t carry meaningful information for classification.
  - **Punctuation and numbers Removal:** Removes common punctuation characters like “,”, “!”, “_” and numbers.
  - **Cleaning:** Eliminates punctuation, numbers, and other non-letter characters to reduce noise.
  - **Lemmatization:** Converts words to their base or dictionary form (e.g., “running” → “run”) to treat similar words the same.
  - **Stemming:** Reduces words to their root form (e.g., “played” → “play”), further simplifying the vocabulary.
- We added the headers “label” and “text.”
- We split the data in X and y for the train and validation data.



MODEL METHODS

SELECTED MODEL

- **Final Model:**
- TF-IDF + NAIVE BAYES
- **Training accuracy:** 0.94%
- **Validation accuracy:** 0.93%
- **No Overfitting !**




```
Model (Naive Bayes + TF-IDF) Train accuracy: 0.9418396105559825
Model (Naive Bayes + TF-IDF) Validation accuracy: 0.9307568438003221

Classification Report:
              precision    recall  f1-score      support
0              0.926768  0.939687  0.933183  3515.000000
1              0.935109  0.921291  0.928148  3316.000000
accuracy       0.930757  0.930757  0.930757     0.930757
macro avg      0.930938  0.930489  0.930665  6831.000000
weighted avg   0.930817  0.930757  0.930739  6831.000000
```

- **Why we selected Model 1 and Model 4?**

Selected models: Model 1 (CV + NB) and Model 4 (TFIDF + NB) were preferred because they show strong and stable validation performance while avoiding obvious overfitting.

- **Why not Model 2 & Model 3?**
- Both Random Forest variants achieved perfect (or near-perfect) training accuracy (1.0) while their validation accuracy dropped — a clear sign of overfitting to the training set.
- Because Random Forests (as configured) memorized the training data more than they generalized to unseen validation data, they were not chosen as the final models.

- **Why NB models were preferred?**

Naive Bayes with bag-of-words or TF-IDF often generalizes better for short headline text and is less prone to overfitting in this setting. They also provide faster training and interpretable behaviour.

CHALLENGES

TAKEAWAYS

Insights:

- TF-IDF highlights key discriminative words.
- Naive Bayes efficiently handles sparse textual data.
- Combining both yields optimal performance with low complexity.

Lessons Learned:

- Proper preprocessing greatly improves text model accuracy.
- Simpler models can outperform complex ones when tuned correctly.

Next Steps:

- Test on larger, real-world datasets.
- Integrate model into a live news verification tool.
- Explore transformer-based models (BERT, RoBERTa) for potential gains

SUMMARY

- TF-IDF highlights the most informative words, improving how the model distinguishes between real and fake news.
- Naive Bayes efficiently models word probabilities and works well with sparse, high-dimensional TF-IDF features.
- The combination achieves strong generalization (0.93 validation accuracy) while avoiding overfitting seen in Random Forest models.

THANKYOU