Fake News Detection

**Objective**

The objective of this assignment is to develop a Semantic Classification model. You will be using Word2Vec method to extract the semantic relations from the text and develop a basic understanding of how to train supervised models to categorise text based on its meaning, rather than just syntax. You will explore how this technique is used in situations where understanding textual meaning plays a critical role in making accurate and efficient decisions.

**Business Objective**

The spread of fake news has become a significant challenge in today’s digital world. With the massive volume of news articles published daily, it’s becoming harder to distinguish between credible and misleading information. This creates a need for systems that can automatically classify news articles as true or fake, helping to reduce misinformation and protect public trust.

In this assignment, you will develop a Semantic Classification model that uses the Word2Vec method to detect recurring patterns and themes in news articles. Using supervised learning models, the goal is to build a system that classifies news articles as either fake or true.

**Pipelines that are performed**

The following tasks are performed:

1. Data Preparation
2. Text Preprocessing
3. Train Validation Split
4. EDA on Training Data
5. Feature Extraction
6. Model Training and Evaluation

**Data Preparation:**

* Data is in two datasets, True.csv and Fake.csv. Both datasets contain three columns:
* title of the news article
* text of the news article
* date of article publication
* New column 'news\_label' is added to both data sets. True news has label is assigned as 1 and for Fake news the label 0 is assigned.
* Both True and Fake news data frames are merged for further processing
* Null Values: Checked for null values and dropped them
* New column “news\_text” is added by joining ‘title’ and ‘text’ and then deleted the columns 'title', 'text', 'date' as they are no longer needed

**Text Preprocessing:**

* Created a new DataFrame to store the processed data and that will have only the cleaned news text and the lemmatized news text with POS tags removed
* Cleaned the text and removed all the unnecessary elements by performing the steps below
  + Made the text lowercase
  + Removed text in square brackets
  + Removed punctuation
  + Removed words containing numbers
  + Removed extra whitespace
* POS Tagging and Lemmatization function is applied for POS tagging and lemmatization, filtering stopwords and keeping only NN and NNS tags
* Stored the cleaned text and the lemmatized text with POS tags removed in separate columns within the new DataFrame

**Train-Validation Split:**

* Dataset is split into 70% train and 30% validation and used stratification on the target variable
* Dataset was split 70:30 into training (train\_df) and validation (val\_df).

**Exploratory Data Analysis (EDA) on Training Data:**

Performed EDA on cleaned and preprocessed texts to get familiar with the training data by performing the tasks given below:

* **Character Lengths:** Plot Histogram to visualise the training data according to the character length of cleaned news text and lemmatized news text with POS tags removed

A graph with a line graph

AI-generated content may be incorrect.

* + **Text Length and Preprocessing Insights:**
    - Text length distribution in both raw cleaned texts and lemmatized texts is heavily right-skewed with most documents falling between 400 and 2500 characters.
    - Lemmatized texts tend to be shorter than clean texts, especially in the lower and mid-length ranges. This is due to lemmatization reducing inflections and redundancies (e.g., "running" → "run").
    - After about 3000 characters, the frequency of longer documents sharply declines, with very few documents exceeding 6000 characters.
* **Top 40 word:** Using a word cloud, found the top 40 words by frequency in true and fake news separately

A close up of words

AI-generated content may be incorrect.

A close-up of words

AI-generated content may be incorrect.

* **n-grams:** Found the top unigrams, bigrams and trigrams by frequency in true and fake news separately
  + **Unigrams:**

**A graph showing different colored bars

AI-generated content may be incorrect.**

**A graph with different colored bars

AI-generated content may be incorrect.**

* + **Bigrams:**

**A chart with different colors

AI-generated content may be incorrect.**

**A graph with different colored bars

AI-generated content may be incorrect.**

* + **Trigrams:**

A graph showing different colored bars

AI-generated content may be incorrect.

A graph showing different colored bars

AI-generated content may be incorrect.

**Feature Extraction**

* To perform classification on textual data for ML model, converted text to a vector form by using the Word2Vec Vectorizer to create vectors from textual data. Word2Vec model captures the semantic relationship between words.

**Model Training and Evaluation**

A set of supervised models are used to classify the news into true or fake.

1. **Logistic Regression Model:** Created and trained logistic regression model on training data

Logistic Regression Evaluation:

**Accuracy:** 93.21%

**Precision:** 92.32%

**Recall:** 93.56%

**F1 Score:** 92.93%

* The model performs very well overall.
* High precision means few false positives.
* High recall indicates few false negatives.
* Balanced F1 score confirms strong performance on both precision and recall.

Classification Report:

|  |  |  |  |
| --- | --- | --- | --- |
| **Metric** | **Class 0 (Fake)** | **Class 1 (True)** | **Overall** |
| **Precision** | 94% | 92% | 93% (macro) |
| **Recall** | 93% | 94% | 93% (macro) |
| **F1 Score** | 93% | 93% | 93% (macro) |
| **Accuracy** |  |  | **93%** |

* The model performs consistently across both classes.
* Class 0 (Fake News): Slightly higher precision.
* Class 1 (True News): Slightly higher recall.
* The F1 score is balanced for both classes, indicating a good trade-off between precision and recall.
* High accuracy (93%) shows strong overall performance.

1. **Decision Tree Model:** Created and trained a decision tree model on training data

Decision Tree Evaluation:

**Accuracy:** 84.53%

**Precision:** 85.03%

**Recall:** 82.01%

**F1 Score:** 83.49%

* The model is relatively accurate when it predicts a positive (true) class.
* Slightly lower recall. It misses some true positives, suggesting it's a bit more conservative in classifying items as positive.
* Balanced F1 score shows reasonable trade-off, but performance is noticeably lower than Logistic Regression.

Decision Tree Classification Report:

|  |  |  |  |
| --- | --- | --- | --- |
| **Metric** | **Class 0 (Fake)** | **Class 1 (True)** | **Overall (Macro Avg)** |
| **Precision** | 84% | 85% | 85% |
| **Recall** | 87% | 82% | 84% |
| **F1 Score** | 85% | 83% | 84% |
| **Accuracy** |  |  | **85%** |

* The model is slightly better at detecting fake news (Class 0) with higher recall (87%).
* Precision is similar for both classes, indicating balanced correctness in positive predictions.
* Overall, performance is solid but less consistent than Logistic Regression.

1. **Random Forest Model:** Created and trained a Random Forest Model on training data

Evaluation:

**Accuracy:** 92.92%

**Precision:** 93.54%

**Recall:** 91.47%

**F1 Score:** 92.49%

* Very high precision
* Slightly lower recall than Logistic Regression → may miss a few true positives.
* F1 score is strong, showing a good balance of precision and recall.
* Overall, Random Forest performs almost on par with Logistic Regression, slightly better in precision, slightly lower in recall.

Random Forest Classification Report Summary:

|  |  |  |  |
| --- | --- | --- | --- |
| **Metric** | **Class 0 (Fake)** | **Class 1 (True)** | **Macro Avg (Overall)** |
| **Precision** | 92% | **94%** | 93% |
| **Recall** | **94%** | 91% | 93% |
| **F1 Score** | 93% | 92% | 93% |
| **Accuracy** |  |  | **93%** |

* The model is very balanced across both classes.
* Slightly stronger at detecting True News (Class 1) in terms of precision.
* Better at detecting Fake News (Class 0) in terms of recall.
* Overall, the model achieves a strong and stable 93% performance across all major metrics.

**Conclusion**

* **Patterns Observed in True vs Fake News:**
  + - * Both true and fake news articles mostly consist of medium-length texts, which aligns with typical news article lengths.
      * Lemmatization likely helps the classifier focus on core semantic content by stripping morphological variation.
      * The right-skewed length distribution and preprocessing make the dataset suitable for text-based semantic classification without heavy bias from extremely long documents.
* **Model Performance Overview:**

A number of percents and percentages

AI-generated content may be incorrect.

* **Semantic Classification Impact:**
  + - * Semantic classification using lemmatized text effectively reduces noise and redundancy, improving model focus on meaningful word stems and concepts, which is critical in distinguishing subtle differences between fake and true news text.
      * The models trained on lemmatized texts achieved high accuracy and balanced metrics, demonstrating that semantic normalization enhances classification quality.
      * Both Logistic Regression and Random Forest models leveraged this preprocessing step to deliver robust and reliable fake news detection.
* **Best Model Choice and Evaluation Metric:**

**Chosen Model: Logistic Regression**

* It offers the best recall (93.56%) and a balanced F1 score (92.93%), ensuring that most fake news instances are detected while maintaining low false positive rates.
* In fake news detection, recall is often prioritized to minimize the risk of letting fake news slip through undetected.

**Alternative Option: Random Forest**

* If minimizing false positives is more important, Random Forest is an excellent alternative due to its higher precision.
* **Assessment and Impact**
  + - * The combined approach of text preprocessing (cleaning + lemmatization) and semantic classification enables effective detection of fake news, with over 93% accuracy from top models.
      * This approach balances the need for sensitivity (recall) and specificity (precision), crucial for real-world applications in media and social platforms.
      * Logistic Regression’s interpretability and solid performance make it a practical choice for deployment in monitoring and flagging fake news content.