Fake News Detection

Build a system to identify unreliable news articles.

G3-CS (CS4824)

Anubhav Nanda
Humaid Desai
Priyanka Savla

Agenda

- Problem Motivation
- Dataset
- Data Transformation
- Machine Learning Techniques
- Tuning
- Evaluation

Problem Motivation

- Increase in usage of the World Wide Web and adoption of social media platforms have led to information dissemination.
- Consumers are sharing more information on social media than ever before, some of which are misleading.
- Partisans have weaponized the fake news to cast aspersions on the veracity of claims made by those who are politically opposed to them.
- There have been real world consequences of fake news like a recipe claimed that garlic cured Coronavirus, shooting at a Pizza Restaurant (PizzaGate - Washington DC) etc.

Dataset

- Kaggle is the source of news dataset which has been described as follows:
- A training dataset with attributes id, title, author, text and a label that marks the article as potentially unreliable where 0 is reliable and 1 is unreliable.
- We split the data into 3 parts using SciKit Learn's train_test_split().
- Ratio of split was
  60%: For training.
  20%: For validation.
  20%: For testing.

Data Transformation

Removal of Blank or missing entries: Initially, we removed rows from the data that contained blank or missing entries.

Converting Text to Lowercase: Transformed the text to lowercase in order to map words with different cases to lowercase form since python is case sensitive.

Word Tokenization: To split news text into a set of words to do further analysis on words.

Data Transformation

Removal of stop words and non-alphabetical words: Helps remove the low-level information text from our dataset and draw more focus on important information.

Lemmatization: Normalize the words to converts them into their meaningful base form using part-of-speech i.e. POS tag.

TF-IDF Vectorization: The TF-IDF value increases proportionally to the number of times a word appears in the document and is offset by the number of documents in the corpus that contain the word.

Machine Learning Techniques

Naive Bayes

- Naive Bayes is a supervised learning algorithm which applies Bayes' theorem with the 'naive' assumption of conditional independence between every pair of features given the value of the class variable.

- We used **MultinomialNB**, a classic naive Bayes variant used in text classification which is known to work well with tf-idf vectors. The model we used correctly classified the genuineness of news with an accuracy of **87.11%**.

Machine Learning Techniques

Support Vector Machine

- For our model, we have considered parameters like Kernel with values {linear, poly, rbf, sigmoid}, Gamma which takes values {scale, auto} and C values ranging from 1 to 5.
- The F1 score is highest for the below parameters which is **94.12**, while lowest F1 score is **66.66**.

	Kernel	C (Regularisation)	Gamma
MAX	RBF	3.0	Scale
MIN	Poly (Deg = 2)	1.0	Auto

Machine Learning Techniques

Logistic Regression

- For our model, we have considered parameters like Solver with values {newton-cg, lbfgs, liblinear, sag, saga}, Penalty which takes {L1,L2}, C ranging from 1 to 5 and Maximum Iterations with values {250, 500, 1000, 1500, 2500}.

- The maximum F1 score is **93.60** and minimum is **91.75** for the below parameters

	Solver	Penalty	C (Regularisation)	Maximum Iterations
MAX	Liblinear	L1	2.0	250
MIN	Newton-cg	none	1.0	250

Tuning

At the time of pre-processing data, we initially used the Stemming technique to change the words into their base form and observed that the classification accuracy was low.

To improve the accuracy, we used a Lemmatizing technique which preserves the context of the words. This technique resulted in **6%** increase in the overall accuracy.

Also, we performed hyperparameter tuning to determine the right combination of hyperparameters that allows the model to maximize model performance.

Evaluation

- **Accuracy -** In the binary news classification problem, the confusion matrix has 2 rows and 2 columns where the rows represent actual values and the columns represent predicted values.
- **Precision -** It is the number of actual reliable news divided by the total number of reliable news predicted.
- **Recall -** It is the number of actual reliable news divided by the total number of reliable news in the test data.
- **F1 Score -** Used to convey the balance between the precision and the recall.

$$F1Score = \frac{2 * precision * recall}{precision + recall}$$

Evaluation

	Naive Bayes	SVM	Logistic Regression
Accuracy	87.66	94.32	93.19
Precision	83.71	94.10	93.29
Recall	90.99	94.56	93.16
F1 Score	87.20	94.33	93.22

Thank You!