Fake News Detection

Machine Learning Project

Made by: Vatshayan
vatshayan007@gmail.com


Fake News
news
media
misinformation
press
journalism
political
fraud
spin
hoax
attacks
newspaper
live coverage
Fake News

Introduction

- Data has been increasing at an unprecedented range in an exponential manner and is producing 2.7 quintillion bytes of data everyday.
- The definition of fake news is information that pushes people down the wrong road. Fake news is spreading like wildfire these days, and people are sharing it without confirming it. This is frequently done to promote or impose specific views, and it is frequently accomplished through political agendas.
- As a result, it is vital to recognise phoney news.

Problem Definition

- Fake News have become more prevalent in recent years and with great amount of dynamism in internet and social media, differentiating between facts and opinions, relating to commercial or political upheavals has become more difficult than ever.

- Fake information is purposely or unintentionally spread throughout the internet. The massive dissemination of fake news has left an indelible mark on people and culture.

- We use various NLP and preprocessing methodologies like tokenization, stop words removal, lemmatization, stemming and machine learning classification algorithms - logistic regression, pac, ada, naive bayes, svm, random forest, xgboost, decision trees and rnn, to build a model that differentiates between fake news and real news and also analyze the performance of these various classification methodologies to choose the best classifier on our dataset.

Types of Data in Social Media Posts

There are three major ways by which social media networking sites read news items:

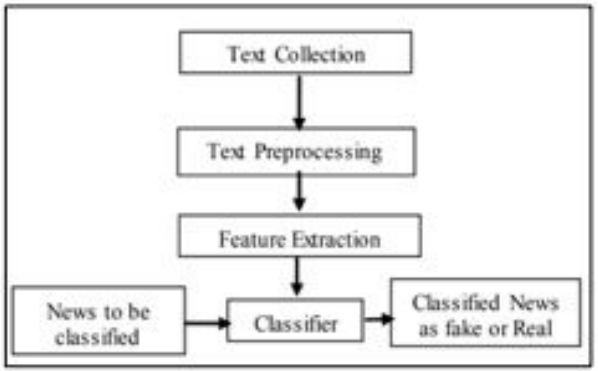
1. **Text:** Computational linguistics analyzes text , focusing on the genesis of text semantically and methodically. Because many of the posts are written in the form of texts, much work has been carried out into analysing them.
2. **Multimedia:** Several types of media are combined in a single post. Audio, video, photos, and graphics may all be included. This is highly appealing, because it captures the attention of the visitors without requiring them to read the content.
3. **Hyperlinks:** Hyperlinks allow the post's creator to cross-reference to other sources, gaining viewers' trust by confirming the post's genesis. Cross-reference to other social media networking sites, as well as the embedding of photos, is common practise.

Types of Fake News and patterns that help in detection

1. **Visual Based:** These false news posts make extensive use of graphics as compared to content, which may include manipulated photographs, doctored video, or a combination of the two.
2. **User Generated News:** This sort of falsified news is generated by phoney accounts and is targeted to certain audiences, which might reflect specific age groups, gender, culture, or political affiliations.
3. **Knowledge based:** These posts provide scientific (so-called) explanations to some unresolved problems, leading people to feel they are genuine. For example, natural therapies for high blood sugar levels in the human body.
4. **Style based:** Pseudo Journalists who impersonate and mimic the style of some accredited journalists write style-based posts.
5. **Stance based:** It is a portrayal of true statements in such a way that its meaning and purpose are altered.

Methodology

The fake news model detection is built using steps like Text Collection, Text Preprocessing, Feature Extraction and then finally classification using different classifiers.



Text Collection

The text collection process is carried out by referring to Kaggle ISOT Fake News datasets. Data is gathered from 244 websites.

It is made up of approximately 44898 posts that were recorded over the course of 30 days. The true news data set is made of 23481 posts and the fake news data set is made of 21417.

The title and text are among the features (news body), the subject, date, and label are all required. The news topics are divided into several categories, including 'politicsNews,' 'worldnews,' 'News,' and 'political', 'Government News,' 'Left News,' 'US News,' 'Middle East.'

Text Pre-processing

Text data must be preprocessed before being input into machine learning and deep learning models, by employing NLP methods such as stop word removal, tokenization, sentence segmentation, and punctuation removal.

Following the acquisition of content, pre-processing is performed.

- All of the letters in the document are converted to lowercase.
- Numbers are removed
- Punctuation and accent marks are removed.
- White spaces are removed
- Stop words are removed

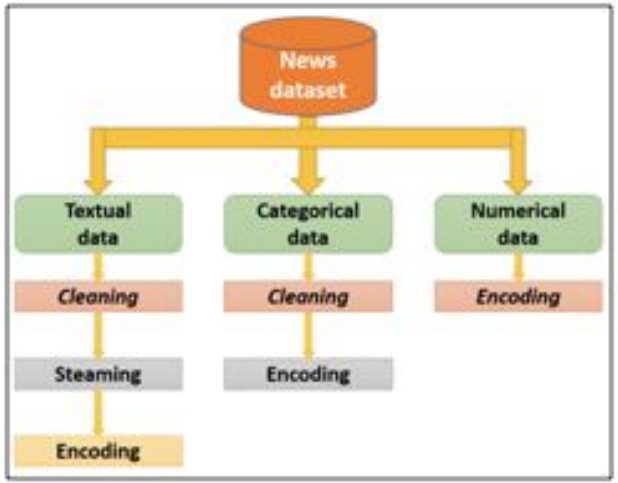
Sample text with Stop Words	Without Stop Words
GeeksforGeeks – A Computer Science Portal for Geeks	GeeksforGeeks , Computer Science, Portal ,Geeks
Can listening be exhausting?	Listening, Exhausting
I like reading, so I read	Like, Reading, read

In-depth text pre-processing

- **Textual data**: Text written by the author in a news, pre-processed by the following operations:
  1. Cleaning: eliminating stop words and special characters.
  2. Stemming: transforming the useful words into roots.
  3. Encoding: transforming all the words into a numerical vector. This needs two steps: bag of words and N-grams, then the application of the TF-IDF method on the result.

- **Categorical data**: Source of the news- TV channel, newspaper or magazine; its author. The pre-treatment of these data is performed through two steps:
  – Cleaning: eliminating special characters and transforming letters into lowercase.
  – Encoding: for sources we used a label encoding. For authors, the names were encoded into numerics. A list containing two fields was created, the first for the source and the second for its authors, then we replaced each author by its index number.

- **Numerical data**: The date of posting and the sentiment given by the text. Split the date into three unique values: day, month and year. For the sentiment given by the text, we calculate the sum of the sentiment degrees of the words.



Classifiers

1. Support Vector Machine:

-One of the most widely used models for binary and multi-classification tasks.

-Supervised machine learning classifier that has been used by numerous academics to solve binary and multi-classification issues.

$$w = \sum_{i=1}^{n} \alpha_i Y_i X_i$$

In a binary classification issue, the instances are separated by a hyperplane in such a way that wTx + b = 0, where w is a dimensional coefficient weight vector that is normal to the hyperplane. The bias term b represents the values offset from the origin, while data points are represented by x. The fundamental job of SVM is to determine the values of w and b.

Drawing decision boundaries is known as creating a hyperplane that separates two classes. Unoptimized decision boundaries may result in misclassification; to overcome this, SVM are regarded as important by examining extreme cases.

2. Logistic Regression:

- popular classification approach in machine learning to predict the values of the predictive variable y in a binary classification issue, where y = [0, 1].
- The negative class is represented by 0 and the positive class by 1.

- To classify two classes, 0 and 1, a hypothesis $h(\Theta) = \theta^T X$ will be constructed, and the classifier's output threshold is when h(x) = 0.5.

- If the value of hypothesis $h(x) \geq 0.5$, it predicts y = 1, indicating that the news is true.

- If the value of hypothesis $h(x) < 0.5$, it predicts y = 0, indicating that the news is false.

- Hence, the prediction of logistic regression under the condition $0 \leq h\theta(x) \leq 1$ is done.

- Logistic regression sigmoid function can be written in equation 5 as follows:

$$h\theta(x) = g((\theta^T X))$$

$$g(z) = 1/(1 + x^{-z}) \text{ and } h\theta(x) = 1/(1 + x^{-\theta^T X})$$

- here:

- Logistic regression Cost function:

$$J(\theta) = \frac{1}{m} \sum_{i=1}^{m} cost(h\theta(x^i), y^i)$$

3. Decision Trees:

- well-known supervised learning algorithm.
- The primary principle behind DT is that it creates a model to forecast the value of a dependent component by learning numerous decision rules derived from the entire set of data.
- Decision Tree features a top-down structure and tree-like shapes, with nodes that can only be a leaf node that is bound with a label class or a decision node that is responsible for making decisions.
- because it is a slow learner, it may perform poorly on small datasets.
- The most important learning process in DT is choosing the right attribute. To tackle this problem, different trees employ different measures, such as information gain in the ID3 algorithm and gain ratio in the C4.5 algorithm.
- For attribute A, the gain ratio and information gain can be calculated as follows:

$$Gain(A, D) = Entropy(D) - \sum_{i=1}^{n} \frac{|D_i|}{|D|} Entropy(D_i)$$

$$GainRatio(A, D) = \frac{Gain(A,D)}{IV(A)}$$

where intrinsic value of attribute A can be calculated as.

$$IV(A) = -\sum_{i=1}^{n} \frac{|D_i|}{|D|} log2 \frac{|D_i|}{|D|}$$

4. Random Forest:

- an ensemble of unpruned decision trees bagged with a randomised set of features at each split. Each tree in the random forest makes a forecast and the forecast with the highest number of votes becomes our final prediction. According to the No Free Lunch theorem, no algorithm is always the most accurate; hence, RF is more accurate and robust than individual classifiers.

The decision tree's high variance was reduced to a low variance by using row sampling and feature sampling. The number of decision trees could be determined using hyperparameters. It's an ensemble algorithm that combines more than one calculation of the same or distinguishing kind for characterizing objects.

The random forest algorithm can be expressed as:

$$F(x) = arg\ max \left\{ \sum_{i=1}^{n} T(A(B, \theta_k)) \right\}$$

If F(x) represents the random forest model, j represents the target category variable, and F represents the characteristic function. To ensure the decision tree's diversity, the sample selection of random forest and the candidate attributes of node splitting are both random.

Algorithm: Random Forest	
Require:	Training set (m is the number of training set, f is the feature set)
Ensure:	Random forest with msub CART trees
1:	Draw Bootstrap sample sets msub with replacement
2:	Choose a sample set as the root node and train in a completely split way
3:	Select fsub randomly from f and choose the best feature to split the node by using minimum principle of Gini impurities
4:	Let the nodes grow to the maximum extent. Label the nodes with a minimum impurity as leaf node
5:	Repeat steps 2-4 until all nodes have been trained or labeled as leaf nodes.
6:	Repeat steps 2-5 until all CART has been trained
7:	Output the random forest with msub CART trees

5. Gradient Boosting

Gradient boosting in machine learning is used for regression and classification. It's a way for boosting. Leaf indicates an initial prediction, which is log(odds) for classification; this is turned into a probability using the logistic function (10).

$$Probability = \frac{e^{log(odd)}}{1+e^{log(odd)}}$$

5. XG-Boost:

It's a very strong Gradient boosting classifier. Designed for usage with large, complex datasets. It is an ensemble strategy that prevents overfitting and thereby regularizes boosting. In all circumstances, it is scalable. It can handle sparse data as well as parallel and distributed processing, making learning faster and more efficient.

7. K-NN:

- K-NN is a well-known machine learning algorithm.
- The K-NN techniques are extremely straightforward.
- Given a test sample, it initially determines the k nearest neighbours based on a distance measure.
- Then, using the major vote strategy, it predicts the class label of the test instance.
- Sometimes the classification performance of K-NN is poor, owing to the curse of dimensionality.
- K-NN is also a lazy learning method that can take a long time to classify data.

Algorithm: KNN Algorithm	
1:	for all unlabeled data u do
2:	for all labeled data v do
3:	compute the distance between u and v
4:	find k smallest distances and locate the corresponding labeled instances v1,.. vk
5:	assign unlabeled data u to the label appearing most frequently in the located labeled instances
6:	end for
7:	end for
8:	End

Summary of steps and procedure

Algorithm	
Input:	News Content
1.	Convert text to lowercase
2.	Remove punctuations, digits, stop words from text
3.	Repeat: Input: Receive each news article Calculate count vector for it Append the count vector to count_feature vector Until the end of news article
4.	Repeat Input: Receive each news article Calculate the TF-IDF vector for it Append the count vector to tfidf_feature vector Until the end of news article
5.	Repeat Input: Receive each news article Calculate the Spacy vector for it Append the spacy vector to Spacy_feature vector Until the end of news article
6.	Parse count_feature vector, tfidf_feature vector and spacy_feature vector into classifier Return feature vector gives us highest accuracy
7.	Build model with the feature vector

Results and Analysis

We can evaluate machine learning algorithms using various metrics like:

1. Accuracy
2. Precision
3. Recall
4. F1-Score

$$Accuracy(Acc)\% = \frac{TP+TN}{TP+TN+FP+FN} \times 100$$

$$Recall(Re)\% = \frac{TP}{TP+FN} \times 100$$

$$Precision(Pre)\% = \frac{TN}{TN+FP} \times 100$$

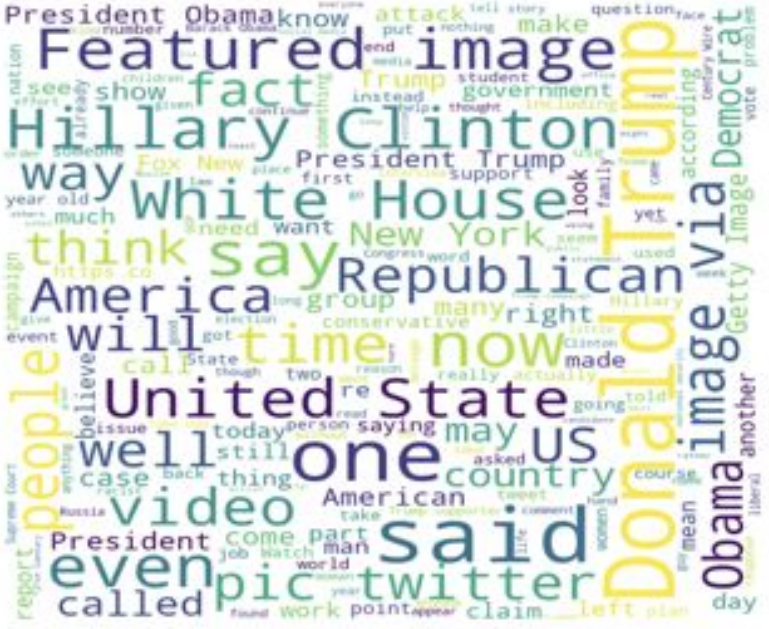
$$F1-Score = 2\frac{(precision)(recall)}{precision+recall}$$

Hence we evaluate and analyse the result based on these metrics for different datasets, classifiers and different methods of feature extraction methodology.

Implementation and results

WordCloud of true news:

WordCloud of fake news:



Conclusion

- The details of our implementation results are given below:

Sr	Models	Accuracy	Precision	F1 Score	Recall
1	Logistic Regression	0.987973	0.986926	0.987387	0.987848
2	ADABoost Classifier	0.988241	0.987661	0.987661	0.987661
3	Passive Aggressive Classifier	0.995367	0.994585	0.995142	0.995700
4	XG Boost	0.990468	0.994342	0.989954	0.985605
5	Random Forest	0.983697	0.986075	0.982838	0.979622
6	Naive Bayes	0.952339	0.951087	0.949930	0.948775
7	SVM	0.994833	0.993287	0.994586	0.995887
8	Decision Tree	0.986726	0.987256	0.986055	0.984857
9	RNN	0.993875	0.993517	0.993632	0.993747

References

[1] N. Smitha and R. Bharath, "Performance Comparison of Machine Learning Classifiers for Fake News Detection," 2020 Second International Conference on Inventive Research in Computing Applications (ICIRCA), 2020, pp. 696-700.

[2] A. Benamira, B. Devillers, E. Lesot, A. K. Ray, M. Saadi and F. D. Malliaros, "Semi-Supervised Learning and Graph Neural Networks for Fake News Detection," 2019 IEEE/ACM International Conference on Advances in Social Networks Analysis and Mining (ASONAM), 2019, pp. 568-569.

[3] S. I. Manzoor, J. Singla and Nikita, "Fake News Detection Using Machine Learning approaches: A Systematic Review," 2019 3rd International Conference on Trends in Electronics and Informatics (ICOEI), 2019, pp. 230-234.

[4] A. Kesarwani, S. S. Chauhan and A. R. Nair, "Fake News Detection on Social Media using K-Nearest Neighbor Classifier," 2020 International Conference on Advances in Computing and Communication Engineering (ICACCE), 2020, pp. 1-4.

[5] K. Poddar, G. B. Amali D. and K. S. Umadevi, "Comparison of Various Machine Learning Models for Accurate Detection of Fake News," 2019 Innovations in Power and Advanced Computing Technologies (i-PACT), 2019, pp. 1-5.


рахмат
danke
謝謝
ngiyabonga
شكراً جزيلاً
Баярлалаа
mersi
kia ora
barka
welalin
spas
tack
teşekkür ederim
mahalo
tapadh leat
спасибо
faafetai lava
vinaka
blagodaram
dank je
misaotra
paldies
grazzi
matondo
спасибі
kiitos
dankie
nanni
nandri
dhanyavad
mauruuru
köszönöm
thank you
gracias
хвала
asante
manana
bedankt
hvala
gràcie
bayarlalaa
akun
dankon
aciu
obrigada
murakoze
tenki
enkosi
djiere dieuf
tau
mochchakkeram
дякую
mamnun
dziękuję
chnorakaloutioun
gratias ago
gràcies
sulpäy
go raibh maith agat
chokrane
sobodi
dêkuji
obrigado
sagolun
sukriya
kop khun krap
taiku
arigatô
takk
dakujem
trugarez
mèsi
didi madloba
ありがとう
tanemirt
grazie
najis tuke
terima kasih
rahmet
ευχαριστώ
diolch
dhanyavadagalu
shukriya
mercé
мерси
kam sah hamnida
rahmat
তোমাকে ধন্যবাদ
감사합니다
xièxie
merci