



# Fake News Classification

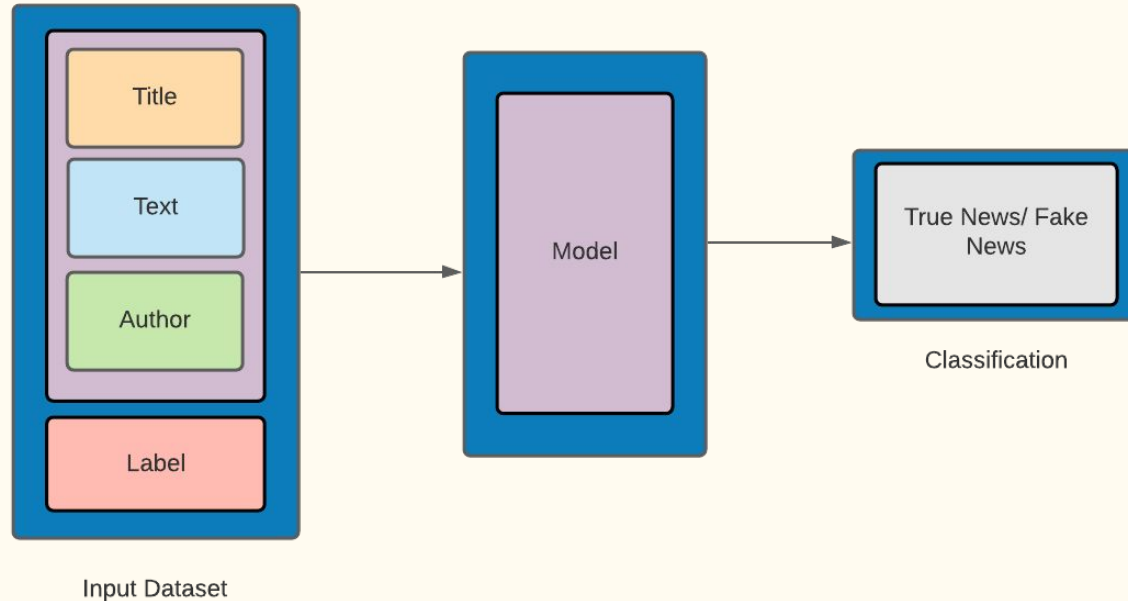
---

Ria Gupta 2018405, Ruhma Mehek Khan 2018362, Prakriti Garg 2019439

# Aim

This project aims to build a fake news classifier that can accurately distinguish fake news from genuine news.

We also aim to develop a simple UI, to enable the users to efficiently verify news articles



# Motivation

- The advent of technology and the development of social media platforms has made it easier to share news updates with the masses.
- The effects of fake news can be catastrophic.
- Fact checking can prevent people from reacting and taking action on fake news.
- Extremely useful for news houses to fact-check their news before they share it with the masses.

# FAKE NEWS

75%

64%

"MAJOR"  
FACBOOK USERS  
BELIEVED IN  
FAKE NEWS

"MINOR"  
BELIEF

BELIEVED IN  
FAKE NEWS  
WITHOUT

Crossed with on  
The issue of fake news

RECENT BUZZFEED SURVEY RESULTS



say fake news has caused \_\_\_\_\_ about  
Americans confused  
events

A great deal  
confusion

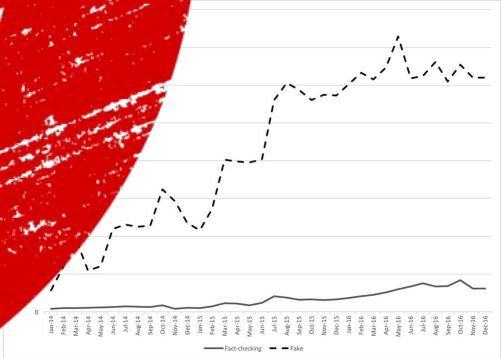
64%

Some  
confusion

24

Not much/  
no confusion

11



# Related Work

- Existing techniques focus more on supervised learning which uses hand crafted input data, which is time consuming. (Kaur,2020)
- Different categories of approaches used for detecting fake news are language approach, topic agnostic, machine learning, and knowledge-based approach.(de Beer,2020)
- More research has been done on Naive Bayes and SVM classifier using Tf-IDF, n-gram features. Models like LSTM, PA, MLP, and features like Hashing vectoriser are less explored.(Kaur,2020)

# Dataset

- ★ 20,800 samples with 10387 real news, 10413 fake news
- ★ Each sample had 5 columns
  - the id of news articles,
  - the title of news articles,
  - news texts,
  - the author names (who reported that article),
  - the label of whether it is classified as fake news or not.
- ★ Can be downloaded from [here](#).

# Exploratory Data Analysis



Figure 3: Wordcloud of all tokenized words

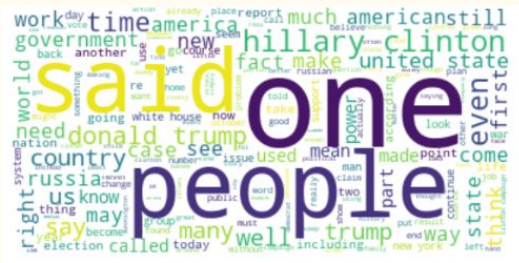


Figure 4: wordcloud of all tokenized words in fake news

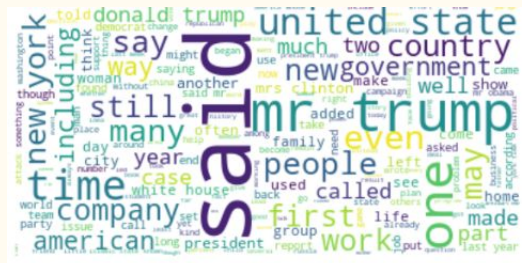


Figure 5: wordcloud of all tokenized words in real news

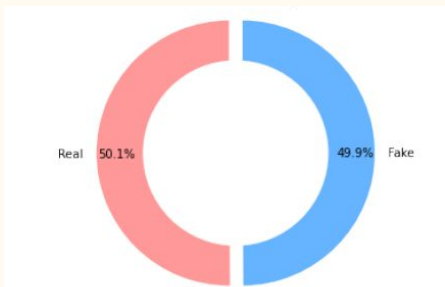


Figure 2: Pie chart of fake and true news

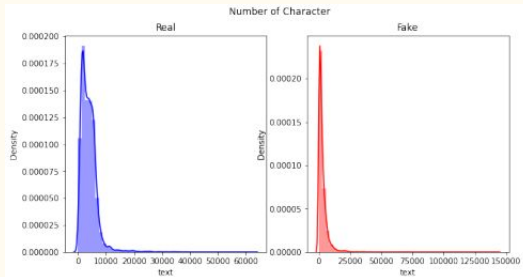


Figure 6: character length plot

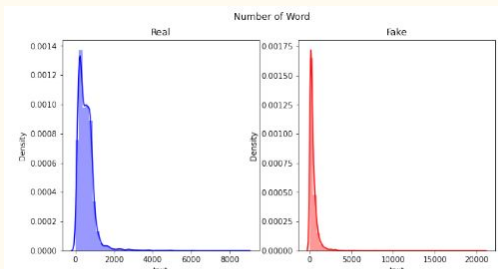


Figure 7: Word length plot

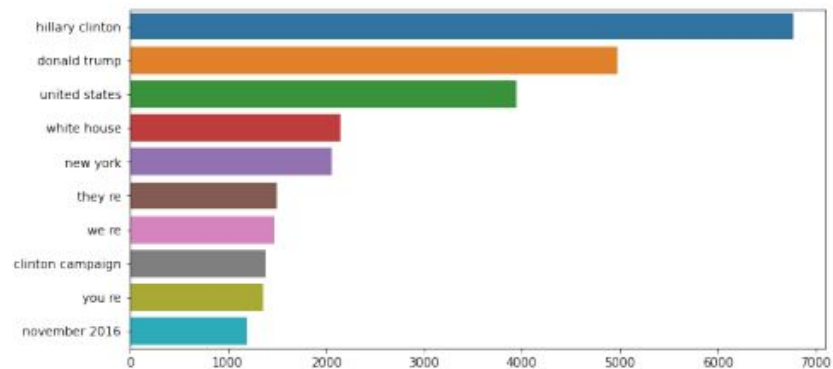


Figure 8: The top 10 bigrams present in fake news

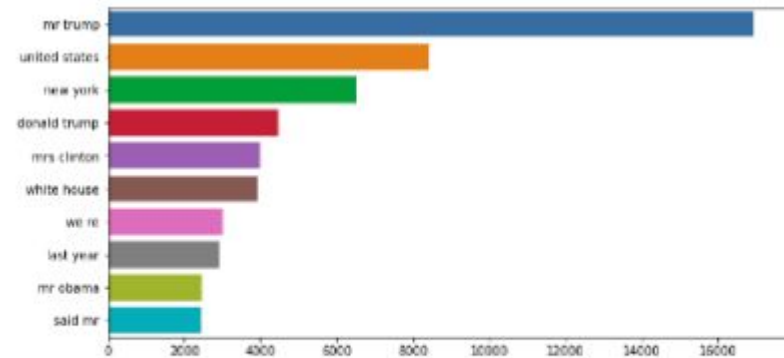
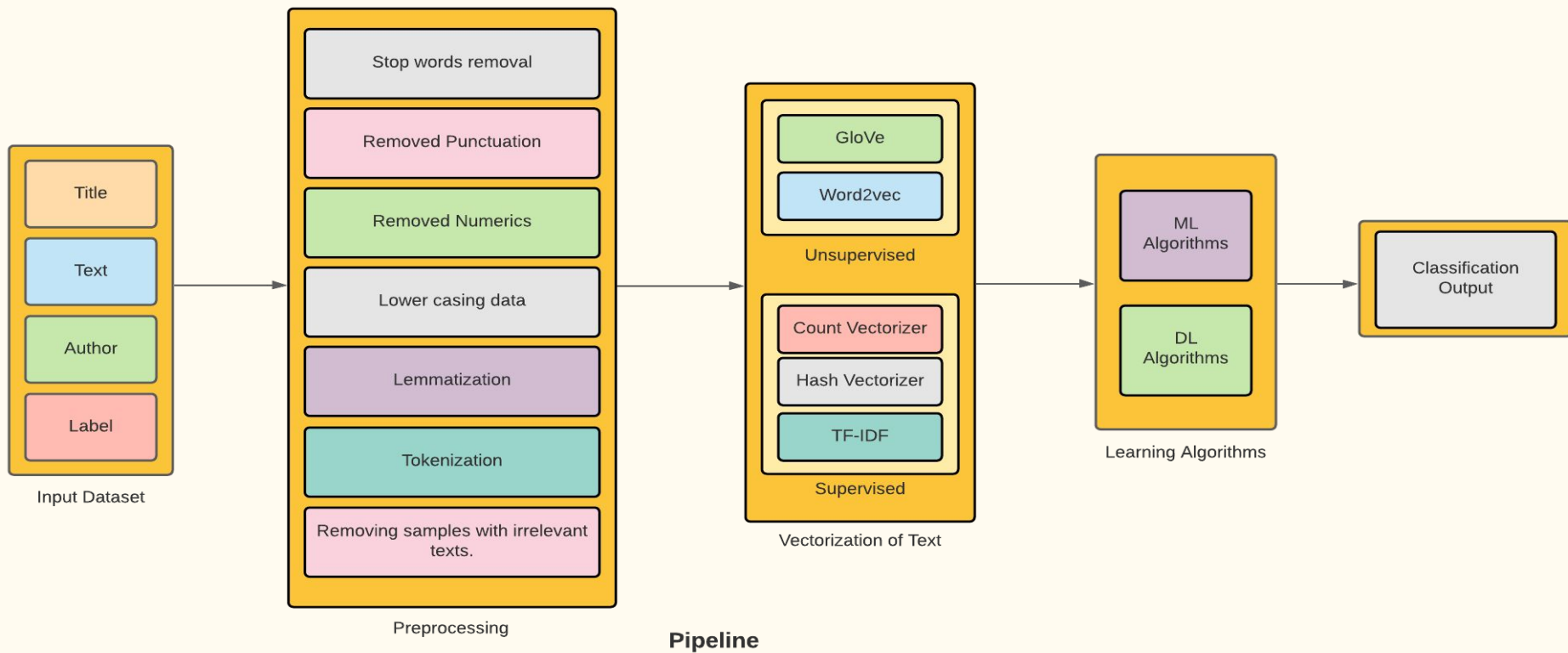


Figure 9: The top 10 bigrams present in true news



# High Level Overview



# Preprocessing

1. Removing irrelevant texts and NAN values
2. Stopwords
3. Numerics and special characters
4. Lemmatization
5. Case folding
6. Tokenization
7. Padding

# Vectorisation

1. OneHot Encoding
2. Count Vectoriser
3. Hashing-Vectorizer
4. TF-IDF
5. GloVe Embedding
6. Word2Vec Embedding
7. BERT

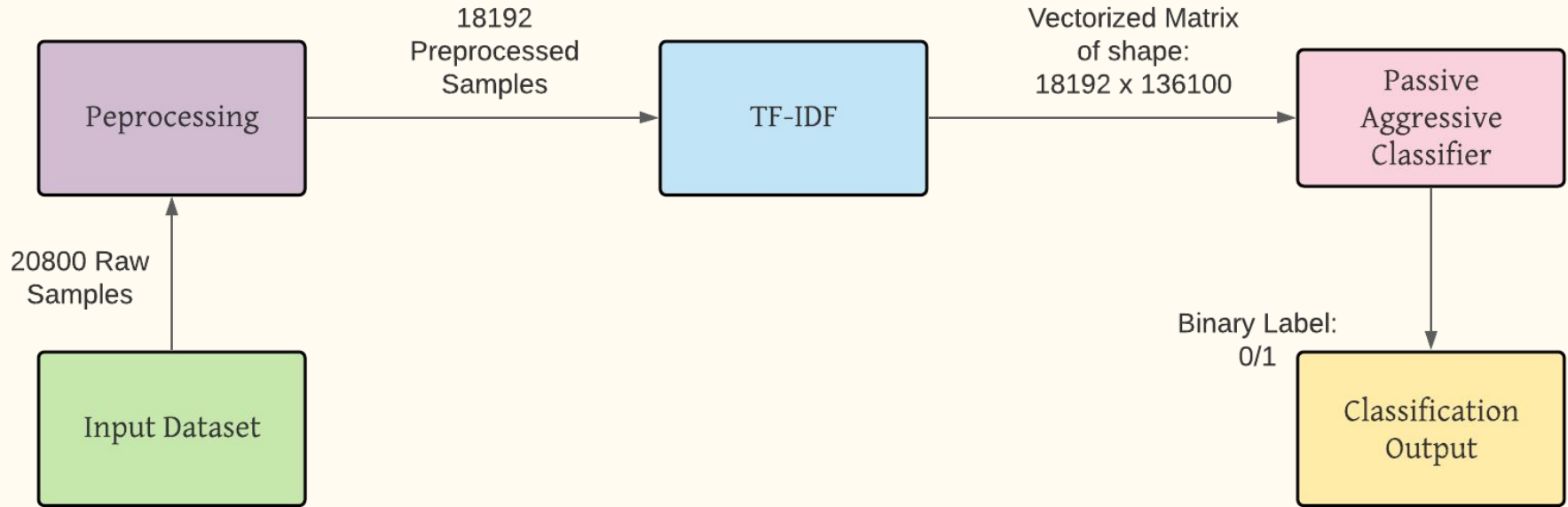
# Machine Learning Algorithms

1. Naive Bayes (MultinomialNB)
2. DecionTree
3. AdaBoost Classification
4. Logistic Regression
5. Passive Aggressive

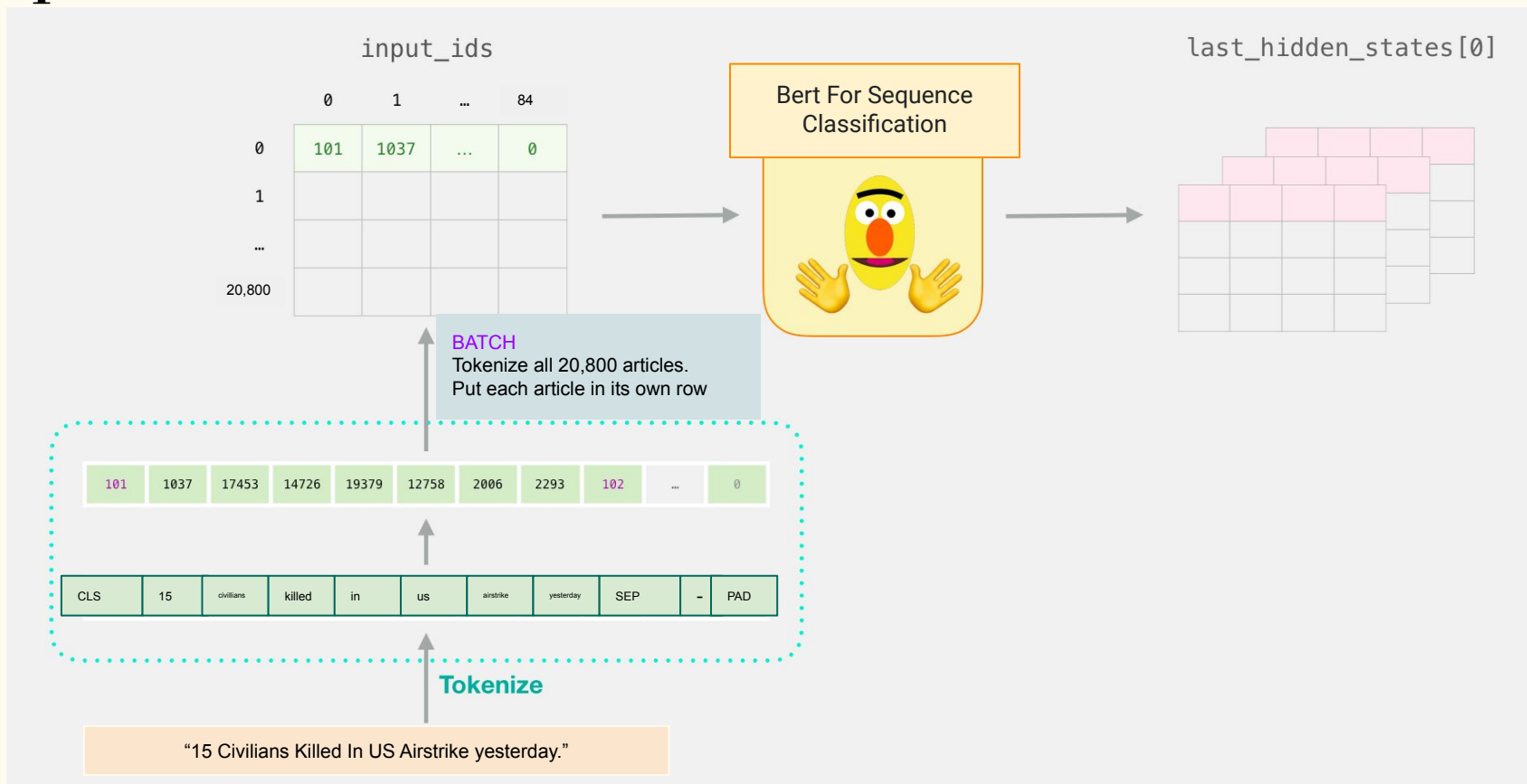
# Deep Learning Algorithms

1. Multilayer Perceptron
2. LSTM
3. BERT

# Pipeline- Passive Aggressive Classifier



# Pipeline - Bert



# Results for BERT Tokeniser with Bert model

Input	Accuracy	Precision	Recall	F1
Text	98.5	99.1	99.1	98.3
Title+Text	98.4	98.8	98.8	98.2
Author+Text	98.6	98.9	98.9	98.6
Text + Title + Author	98.5	99.4	99.4	98.4



# Results

For TF-IDF Tokenizer + Passive Aggressive Classifier

Input	Accuracy	Precision	Recall	F1
Text	96.7	96.3	96.1	96.2
Title+Text	97.2	97.1	96.5	96.8
Author+Text	97.2	98.6	96.8	96.8
Text + Title + Author	97.3	97.1	96.4	96.7

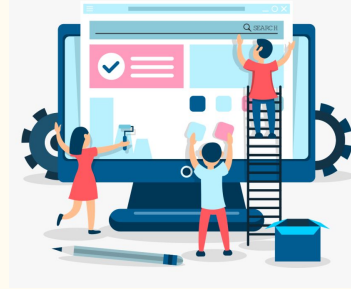
# Summary

- Tf-idf works best for ML models
- Bi-LSTMs with GloVe Embeddings also performed decently well.
- Bert - most accurate (accuracy: 98.5%)

Featurizer	Model Arch	3-fold cv	Testing acc
CV	NB	91.74	92.3
CV	DT	88.78	89.4
CV	LR	95.46	95.6
CV	AdaB	93.07	93.3
CV	MLP	95.79	96.4
CV	GB	93.21	93.7
CV	PA	94.21	95.2
CV	XGB	93.17	93.5
TF-IDF	NB	73.06	77.1
TF-IDF	DT	87.62	88.9
TF-IDF	LR	94.85	95.8
TF-IDF	AdaB	92.99	93.4
TF-IDF	MLP	96.11	96.6
TF-IDF	GB	93.39	82.5
TF-IDF	PA	96.33	96.9
TF-IDF	XGB	93.27	82.0

Table 1: Model performances

# [Bonus]: WebApp



<https://fake-news-detection-nlp.herokuapp.com/>

- A simple minimalistic user interface
- Allows user to input title/text/author
- ‘Load sample input’ button allows users to test app
  - Verifies articles in a click
- Efficient and accurate tool for fact checking

# Future Work



1. Building a chrome extension, for checking fake news on all social media platforms.
2. Adding an option of searching by link, i.e users should be able to add the news link, and the required information is web-scraped from there.
3. Allowing users to verify their prediction, which would then be added in our training dataset- will allow Model to learn overtime.
4. Suggesting similar genuine news.

*Was the prediction correct??*



# Learnings



1. Understood advantages and disadvantages of various preprocessing and vectorisation techniques
2. Hands-on experience with different ML and DL models
3. Understood and implemented Bert
4. Used real world dataset and NLP approaches to solve the problem of Fake News
5. Developed and Deployed a functioning Web App

# Individual Contribution

## 1. PRAKRITI GARG 2019439

Literature Review, Exploratory Data Analysis, Vectorising and applying Machine Learning, Deep Learning Algorithms.

## 2. RUHMA MEHEK KHAN 2018362

Preprocessing data, vectorising and applying machine learning, deep learning models

## 3. RIA GUPTA 2018405

Generating embeddings(GloVe, Word2Vec, BERT), Building a webapp, and deploying on Heroku

# References

- 1.Kaur, S., Kumar, P. and Kumaraguru, P., 2019. Automating fake news detection system using multi-level voting model. *Soft Computing*, 24(12), pp.9049-9069.
- 2.de Beer, D. and Matthee, M., 2020. Approaches to Identify Fake News: A Systematic Literature Review. *Integrated Science in Digital Age 2020*, pp.13-22.