Karunya INSTITUTE OF TECHNOLOGY AND SCIENCES
(Declared as Deemed to be University under Sec.3 of the UGC Act, 1956)
MoE, UGC & AICTE Approved
NAAC A++ Accredited

An internship report submitted by

MANOAH NOBLE (URK21CS1173)
MADHAN L (URK21CS1104)
NATHAN SHIBU JOHN (URK21CS3003)

in partial fulfillment for the award of the degree of

BACHELOR OF TECHNOLOGY
in
COMPUTER SCIENCE AND ENGINEERING

under the supervision of

Dr. T. JEMIMA JEBASEELI

ASSOCIATE PROFESSOR

DIVISION OF COMPUTER SCIENCE AND ENGINEERING
KARUNYA INSTITUTE OF TECHNOLOGY AND SCIENCES
(Declared as Deemed to be University under Sec-3 of the UGC Act, 1956)
Karunya Nagar, Coimbatore - 641 114. INDIA

Karunya INSTITUTE OF TECHNOLOGY AND SCIENCES
(Declared as Deemed to be University under Sec.3 of the UGC Act, 1956)
MoE, UGC & AICTE Approved
NAAC A++ Accredited

DIVISION OF COMPUTER SCIENCE AND ENGINEERING

BONAFIDE CERTIFICATE

This is to certify that the report entitled, "**Fake News Detection using Machine Learning**" is a bonafide record of Internship work done at **Intel Unnati Industrial Training** during the academic year 2022-2023 by

MANOAH NOBLE (URK21CS1173)
MADHAN L (URK21CS1104)
NATHAN SHIBU JOHN (URK21CS3003)

in partial fulfillment of the requirements for the award of the degree of Bachelor of Technology in Computer Science and Engineering of Karunya Institute of Technology and Sciences.

Dr. T. JEMIMA JEBASEELI

ASSOCIATE PROFESSOR

ACKNOWLEDGEMENT

First and foremost, I praise and thank ALMIGTHY GOD whose blessings have bestowed in me the will power and confidence to carry out my internship.

I am grateful to our beloved founders **Late. Dr. D.G.S. Dhinakaran, C.A.I.I.B, Ph.D** and **Dr. Paul Dhinakaran, M.B.A, Ph.D**, for their love and always remembering us in their prayers.

We extend Our tanks to **Dr. Prince Arulraj, M.E., Ph.D.,** our honorable vice chancellor, **Dr. E. J. James, Ph.D.,** and **Dr. Ridling Margaret Waller, Ph.D.,** our honorable Pro-Vice Chancellor(s) and **Dr. R. Elijah Blessing, Ph.D.,** our respected Registrar for giving me this opportunity to do the internship.

I would like to thank **Dr. Ciza Thomas, M.E., Ph.D.,** Dean, School of Engineering and Technology for her direction and invaluable support to complete the same.

I would like to place my heart-felt thanks and gratitude to **Dr. J. Immanuel Johnraja, M.E., Ph.D.,** Head of the Department, Division of Computer Science and Engineering for his encouragement and guidance.

I feel it a pleasure to be indebted to **Dr. T. Jemima Jebaseeli**, Associate Professor, Division of Computer Science and Engineering & **Mr. Nazneen Sultana**, AI Software Solutions Engineer AI & Analytics, Intel Corporation for their invaluable support, advice and encouragement.

I also thank all the staff members of the School of CST for extending their helping hands to make this in Internship a successful one.

I would also like to thank all my friends and my parents who have prayed and helped me during the Internship.

Fake news detection using machine learning

1.0 Introduction

In this section, we will provide an overview of the project, highlighting the problem statement and objectives. Additionally, we will present a chapter wise summary to give a glimpse of the report's structure.

1.1 Problem Statement

The proliferation of fake news has become a significant challenge in today's digital age. With the rapid spread of information through social media and online platforms, it has become increasingly difficult to distinguish between reliable and fabricated news articles. The problem is further exacerbated by the potential impact of fake news on public opinion, decision-making processes, and even political landscapes.

1.2 Objective

The objective of this project is to develop an effective machine learning system capable of accurately detecting fake news articles. By leveraging the power of artificial intelligence and natural language processing techniques, we aim to build a robust and scalable solution that can analyse textual content and identify misleading or false information.

1.3 Apply Intel optimization for enhanced performance

Intel provides optimization tools and libraries that can boost the performance of deep learning models on Intel architectures. In this project, we will explore Intel optimization techniques. In this project we have used the intel tensorflow package which helps us to run the neural network part without necessary of the GPU and run more efficiently and also intel has provided the sklearnex package which 100x faster than sklearn package which is available in the internet.

1.4 Chapter wise Summary

1. **Introduction**: This chapter provides an overview of the project, including the problem statement and objectives.
2. **Exploratory Data Analysis & Visualization of Dataset:** We will explore and analyze the Fake and True news datasets here. We will preprocess the data and visualize it using various techniques.
3. **Model Training and Testing:** This chapter focuses on the training and evaluation of seven different models.
   a) Logistic Regression
   b) Decision Tree Classifier
   c) Random Forest
   d) Support Vector Machine
   e) Gradient Boosting
   f) Neural Networks

   We will provide details about each model's architecture, training process, performance analysis, and the comparison between each model.

4. **Code:** The code is written and executed in python language which will consist of importing the datasets, exploratory data analysis, data preprocessing, data vectorization, model training and testing and finally comparison of accuracy scores of each model.
5. **Conclusion:** The final chapter summarizes the findings of the project, highlights why a model didn’t work out as planned.

2 Exploratory Data Analysis & Visualization of Dataset:

In this section, we will explore the fake news dataset and perform data analysis and visualization. We will outline the datasets characteristics and preprocess the data for the model training and testing.

[61]:

	text	label
0	president 's women duterte 's fiercest critics...	1
1	manufacturers wall street getting trump 's ear...	1
2	trump threatens u.c berkeley protests stop far...	1
3	trump got caught hot mic makes another disgust...	0
4	cubans sold everything reach u.s. hundreds str...	1
...	...	...
44893	kellyanne conway ' husband blasts trump ' trav...	0
44894	teen jackie evancho first singer confirmed tru...	1
44895	update pressure miss usa flip flopped healthca...	0
44896	model outs men moms wives send gross illicit p...	0
44897	president obama burns canadian ted cruz epic j...	0

44898 rows × 2 columns

2.1 Data preprocessing

Data preprocessing is a crucial step in machine learning that involves transforming raw data into a format suitable for training a machine learning model.

This are the steps used to make a suitable format to train the model:

1. Tokenize this process helps to split the text into words and make into a array.

2. lemmatization is to reduce inflectional forms and sometimes derivationally related forms of a word to a common base form. For instance:
   am, are, is $\Rightarrow$ be
   car, cars, car's, cars' $\Rightarrow$ car

3. Stop-words In machine learning and natural language processing (NLP), stop-words refer to commonly used words that are considered to have little or no significance in determining the meaning or sentiment of a text.

   Here are some key points about stop-words in machine learning:
   1) Reducing Dimensionality
   2) Focus on Important Words
   3) Improved Feature Extraction
   4) Language and Context Specificity
   5) Contextual Considerations

2.3 Data Visualization

In this subsection, we provide visual representations of the dataset. Which helps to classify the different types of news by visualizing the dataset. Some of the visualization techniques we employ include:

Bar Graph for types of different news in the data set:

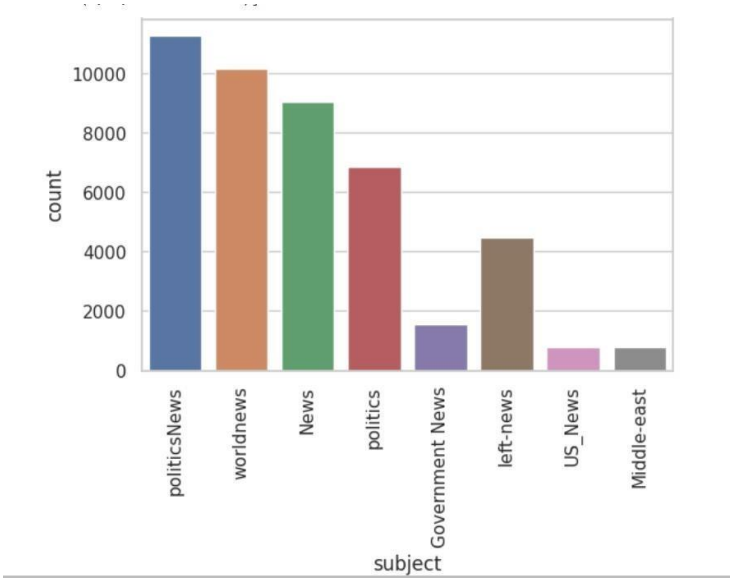

Fig 1 it helps to differentiate various news

Bar Graph for Number of real and fake news:

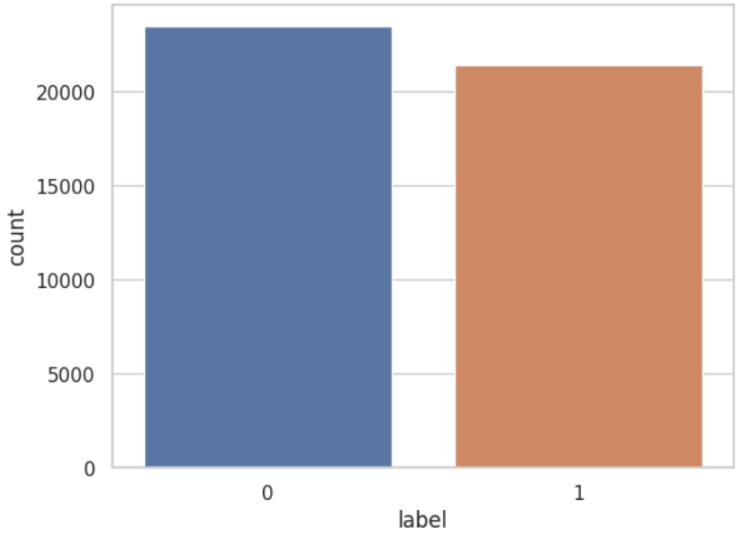

Fig 2 it helps to find the majority of fake news compare to the real news on the data set

3 Models

3.1 Logistic Regression

Logistic regression is a relatively simple and straightforward algorithm. It is based on the concept of linear regression and uses a logistic function (sigmoid) to map the output to a probability between 0 and 1, which suits the requirement of our project.

One of the biggest advantages of logistic regression is its efficiency with small datasets however in our case, this proves as a disadvantage as the combined csv files of fake and true news adds up to over 3000 entries.

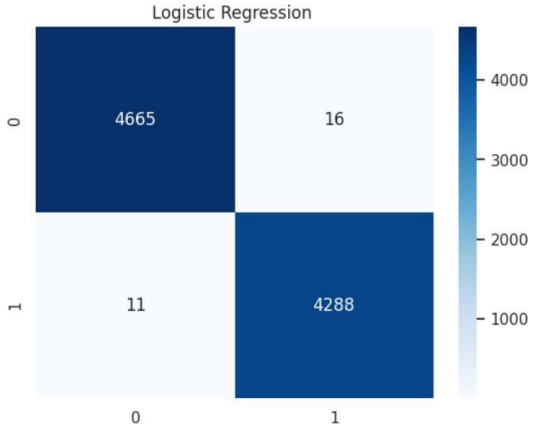

Fig 3 Confusion matrix

3.2 Decision Tree Classifier

Decision trees provide a clear and intuitive representation of the decision-making process. The tree structure can be visualized and easily understood, allowing users to interpret the rules and conditions that lead to different classification outcomes. Also, decision trees can handle irrelevant features or features that have little impact on the target variable more effectively than logistic regression.

However, a similar problem arises as the decision tree classifier model is good with smaller datasets but decision trees can become computationally expensive and memory-intensive as the dataset size increases.

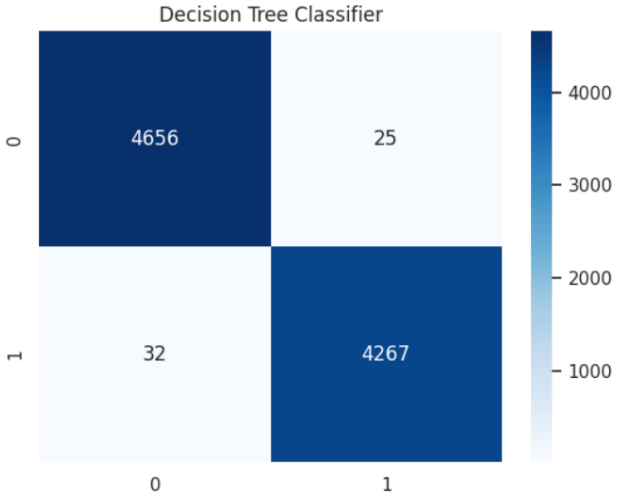

Fig 4 Confusion matrix

3.3 Random Forest

Random Forest often provides higher accuracy compared to a single decision tree. It combines multiple decision trees by averaging or voting their predictions, reducing the impact of individual trees' biases and errors.

Random forests are primarily designed to capture nonlinear relationships between features and target variables. If the underlying relationships in the data are primarily linear, other models like linear regression may perform better.

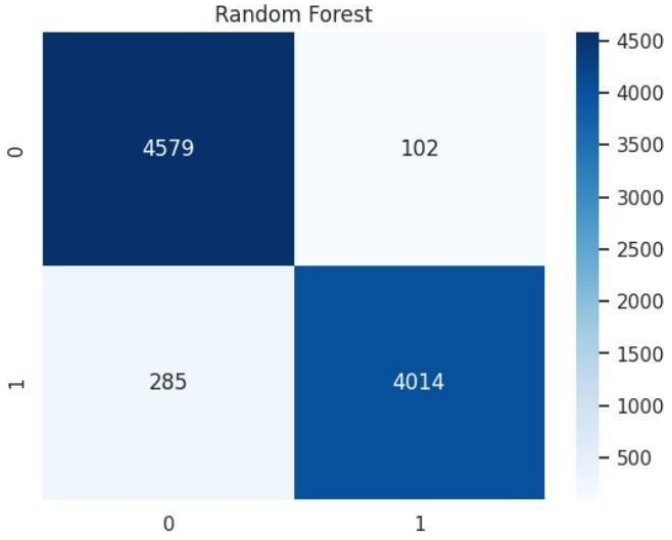

Fig 5 Confusion matrix

3.4 Naive Bayes

Naive Bayes models are commonly used for text classification tasks. However, if the test data contains words or phrases that were not present in the training data (out-of-vocabulary words), the model will assign them zero probabilities, making it challenging to make accurate predictions. Techniques like smoothing or incorporating domain-specific knowledge can help address this problem.

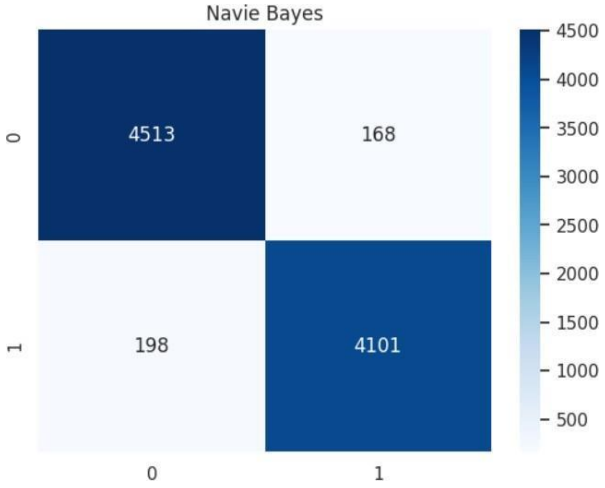

Fig 6 Confusion matrix

3.5 Support Vector Machine

SVMs may face difficulties in high-dimensional datasets, especially when the number of features is much larger than the number of instances. In such cases, the SVM model may have a harder time finding an optimal decision boundary due to the curse of dimensionality.

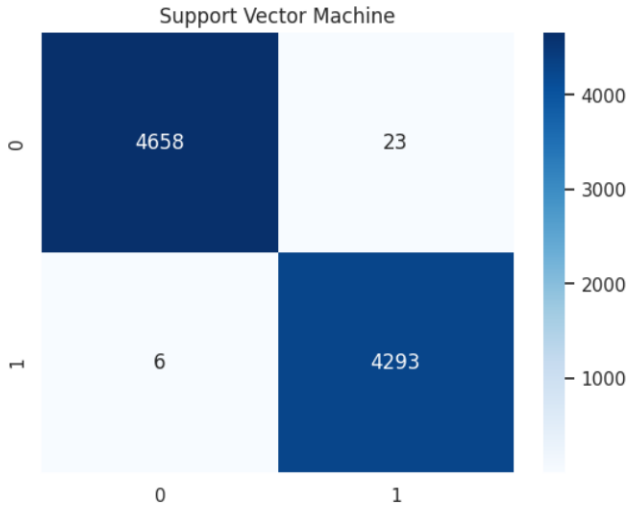

Fig 7 Confusion matrix

3.6 Gradient Boosting

Gradient Boosting models can be computationally expensive and time-consuming, particularly when dealing with large datasets and complex models. Training a deep and wide gradient boosting model with many trees and intricate configurations may require substantial computational resources and time.

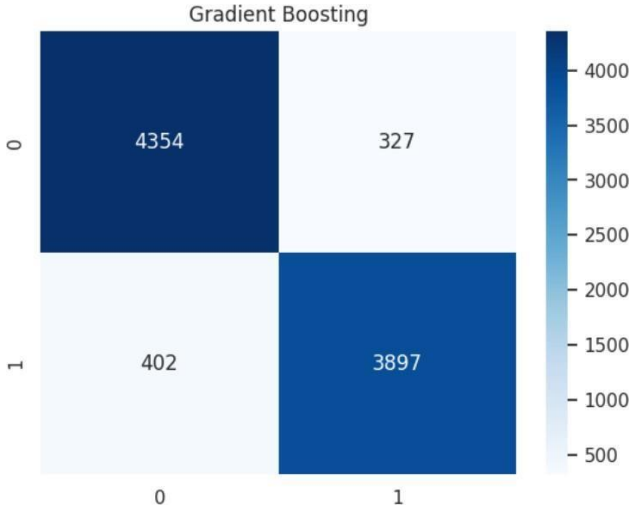

Fig 8 Confusion matrix

3.7 Neural Network

Neural networks have several hyperparameters that need to be tuned carefully to achieve optimal performance. Choosing the right network architecture, activation functions, learning rates, batch sizes, and regularization techniques can be challenging and require extensive experimentation**.**

```
Model: "sequential"
_________________________________________________________________
 Layer (type)                Output Shape              Param #
=================================================================
 Embedding (Embedding)       (None, 5182, 20)          2523520

 global_average_pooling1d (G  (None, 20)               0
 lobalAveragePooling1D)

 dense (Dense)               (None, 20)                420

 dense_1 (Dense)             (None, 1)                 21

=================================================================
Total params: 2,523,961
Trainable params: 2,523,961
Non-trainable params: 0
```

Fig 9 Neural Network layers summary

```
scoreDataSet = pd.DataFrame(dict)
scoreDataSet
```

	LR	NN	GB	SVM	NB	RF	DTC
0	99	98	91	99	96	98	99

Fig 10 Dataset of the model accuracy values

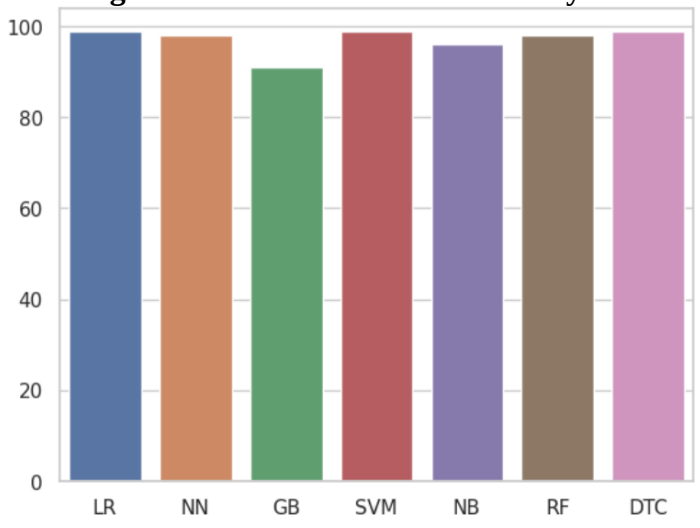

Fig 11 Accuracy score of models

1. Limitation of Gradient Boosting

In this section we will be explaining the reason behind gradient boosting model for less accuracy score compare to the other models.

Out of the 7 models we trained and tested, the Gradient boosting model only provided 92% accuracy which was lesser than what was expected. The reason for the accuracy score to be comparatively low can be:

1) **Insufficient data**: As we know Gradient Boosting is a model that deals with a larger dataset. The limited nature of the dataset must have had an impact on the model's accuracy. Also, the model requires quality datasets that represent real-world scenarios for better accuracy scores.

2) **Model hyperparameters**: Gradient boosting models have various hyperparameters that can affect

their performance. If the hyperparameters are not tuned accurately for the specific dataset and problem, the model might not reach its full potential accuracy.

3) **Inherent challenges in fake news detection**: Fake news detection is a challenging task, as it involves analyzing complex textual information and identifying subtle patterns. Fake news can be designed to closely mimic genuine news, making it difficult even for Gradient Boosting algorithms to distinguish accurately.

4) **Limited interpretability**: Gradient boosting models, particularly when using complex tree-based learners, tend to have limited interpretability. Understanding the specific features or patterns that contribute to the model's decisions can be challenging.

2. Conclusion

In this section, we summarize the findings of our project and provide concluding remarks. We highlight the key points discussed throughout the report and offer insights into the performance and effectiveness of the models trained for fake news detection using Machine learning.

5.1 Summary of thing

We summarize the main finding of our project, including:
The development of seven different models and they are Logistic Regression, Random Forest, Decision Tree Classifier, Sequential Neural Network, Navie Bayes, Support Vector Machine, Gradient Boosting.

Evaluation of the models' accuracies: We analyse the accuracy achieved by each model on the fake news detection test dataset. This evaluation helps us determine the model that performs best for real or fake news classification.

And we use the intel oneApi dev library which helps to run the code faster and more efficient way. By using this library, we develop the model which helps to predict the accuracy for certain model.

3. References

1) About Keras : https://keras.io/guides/sequential_model/
2) Random Forest Model: https://www.ibm.com/topics/random-forest#:~:text=Random%20forest%20is%20a%20commonly,both%20classification%20and%20regression%20problems
3) Navie Bayes: https://www.javatpoint.com/machine-learning-naive-bayes-classifier
4) Logistic Regression: https://www.w3schools.com/python/python_ml_logistic_regression.asp
5) Gradient boosting: https://www.mygreatlearning.com/blog/gradient-boosting/
6) Support Vector Machine: https://www.javatpoint.com/machine-learning-support-vector-machine-algorithm
7) Sequential model by tensorflow: https://www.tensorflow.org/guide/keras/sequential_model
8) Decision Tree Classifier: https://scikit-learn.org/stable/auto_examples/tree/plot_unveil_tree_structure.html#sphx-glr-auto-examples-tree-plot-unveil-tree-structure-py