COMPARATIVE ANALYSIS OF VARIOUS ALGORITHMS FOR FAKE NEWS DETECTION

by

Prithwiraj Samanta *and Satya Kumari*

Under the guidance of

Dr. Rashmi Panda

भारतीय सूचना प्रौद्योगिकी संस्थान राँची

INDIAN INSTITUTE OF INFORMATION TECHNOLOGY, RANCHI

(An Institution of National importance under act of Parliament)

(Ranchi - 834010), Jharkhand

Department of Computer Science and Engineering

Introduction

- Fake news intends to convince the reader to believe false information.
- Published fake news used to increase readership or as part of psychological warfare.
- Fake news detection is the task of detecting forms of news consisting of deliberate disinformation or hoaxes.
- We can prevent lot of negative impact on the society caused by fake news.
- We have used Deep Learning to approach this problem.

Literature Survey

Author	Algorithm	Accuracy
Zhou et al.[1]	Not Applicable	Not Applicable
Abedalla et al.[2]	Attention Layer with CNN + LSTM	Accuracy of 0.71
Fawaid et al.[3]	Fine Tuned BERT	Accuracy of 0.90
Ajao et al.[4]	RNN	Accuracy of 0.82
Manzoor et al.[5]	Random Forest Classifier	Accuracy of 0.65

Literature Survey

Author	Algorithm	Metric(Best)
Zhao et al.[6]	Combination of Clustering and Classifiers on Clusters	Accuracy of 0.33
Ahmed et al.[7]	Passive Aggressive Classifier	Accuracy of 0.93
Umer et al.[8]	CNN + LSTM (with PCA & MSWD)	Accuracy of 0.97
Dong et al.[9]	DSTL	F1-Score of 0.58
Salur et al.[10]	CNN + LSTM in Parallel	F1-Score of 0.89

Motivation

- We came across various algorithms during our research.
- The best performer was CNN + LSTM model. We decided to work further on this model and try to improve its performance.
- During our work we also came across Attention and Transformer based models which are emerging slowly.
- We tried to perform a comparative analysis of CNN + LSTM, Attention, and Transformer models.

Methodology

- RNNs can keep track of arbitrary long-term dependencies in the input sequences.
- When training a RNN using back-propagation, the long-term gradients which are back-propagated can "vanish" or "explode“.
- It is because of the computations involved in the process, which use finite-precision numbers.
- RNNs using LSTM units partially solve the vanishing gradient problem, because LSTM units allow gradients to also flow unchanged.

Methodology

- A common LSTM unit is composed of a cell, an input gate, an output gate and a forget gate.
- CNNs are regularized versions of multilayer perceptron. Multilayer perceptron usually mean fully connected networks.
- CNNs take advantage of the hierarchical pattern in data.
- It assemble patterns of increasing complexity using smaller and simpler patterns embossed in their filters.
- CNN layers used for feature extraction on input data combined with LSTMs to support sequence prediction.

Methodology

- Attention is a mechanism combined in the RNN allowing it to focus on certain parts of the input sequence when predicting a certain part of the output sequence, enabling easier learning and of higher quality.
- Combination of attention mechanisms enabled improved performance in many tasks making it an integral part of modern RNN networks.
- Attention mechanisms let a model draw from the state at any preceding point along the sequence.

Methodology

- The attention layer can access all previous states and weights them according to a learned measure of relevancy, providing relevant information about far-away tokens.
- Transformers use an attention mechanism without an RNN, processing all tokens at the same time and calculating attention weights between them in successive layers.

Implementation

Pre-processing

- The text data which is available to us for using in the fake news detection is full of noisy information and present in format that can’t be directly used by our algorithms.
- So, we have done the data pre-processing to transform the data into a useable form.
- We started with removing the null values present in the dataset. We have dropped the entire row in which any column is null.

Implementation

- We have expanded the contraction(short forms and shorthand's) used in the english text using contractions library of python.
- We converted the text into lowercase and have split the text into word list.
- We have removed all form of punctuation and stop words from the processed word list.
- We have removed all special characters and numbers from the processed word list.
- We have removed all non-English words from the word list and then again convert it back to text.

Implementation

- We have use TF-IDF vectorizer to extract the features from the corpus(processed text) and convert it to word vector.
- We have also created a separate form of dataset using tokenizer instead of vectorizer.
- At last, we save both the format using pickle library of python. We have generated both the format used by common ML algorithms
- Then we used stemming to generate another form of formatted data. We will be using a python library, Port Stemmer.
- We used one-hot encoder to encode the words and then generate the embedding matrix.

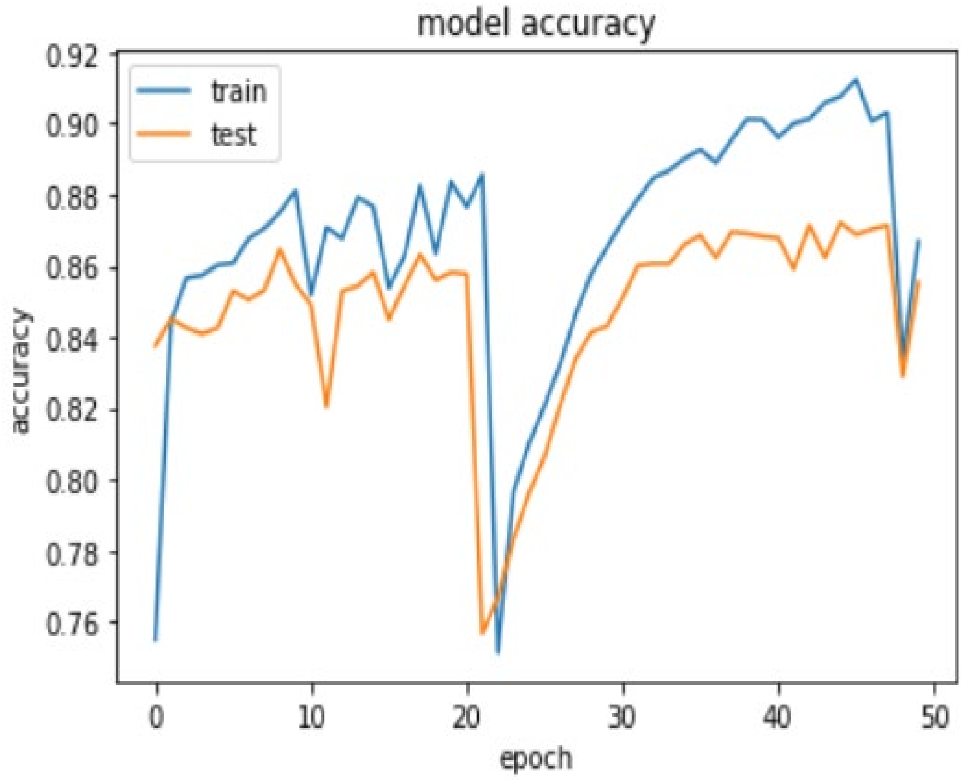
Implementation

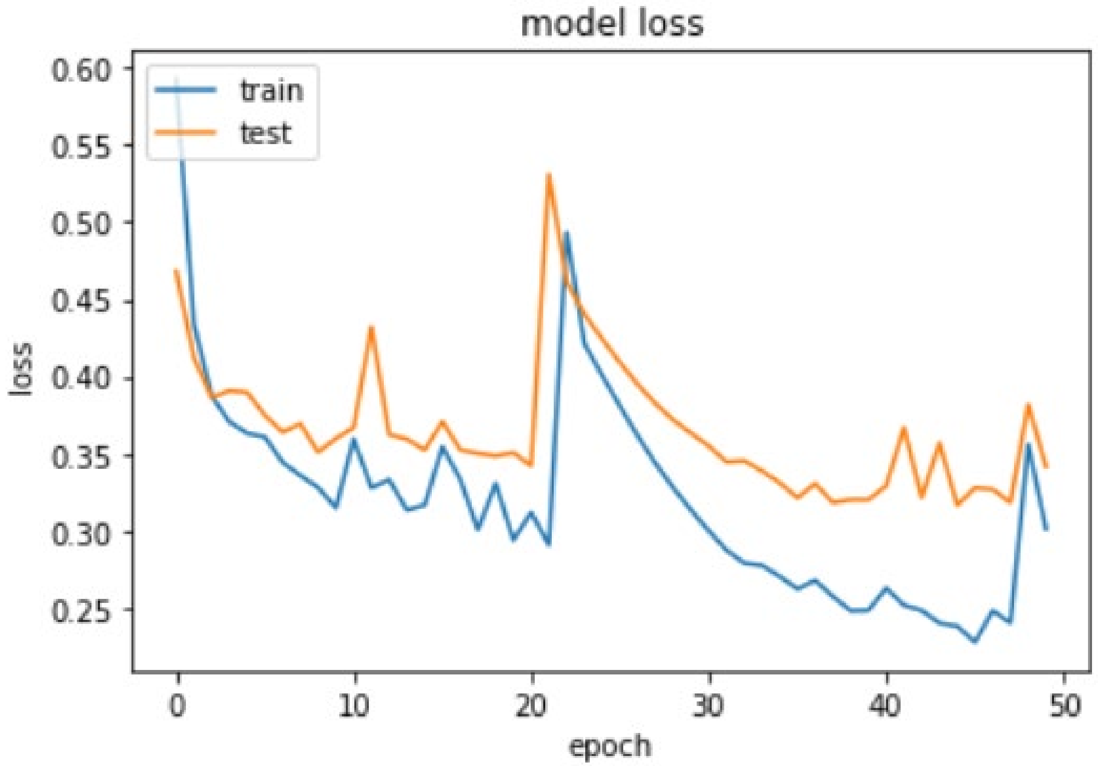
Model-1

- It is a simple LSTM model with 100 features. We observe that its validation accuracy is 86% though we have the training accuracy 2-3% higher.
- There is a good amount of oscillation in the accuracy during training and there is steep change in curve at few points.
- We found that simple LSTM model requires a large number of features to converge.

Implementation

Model-1





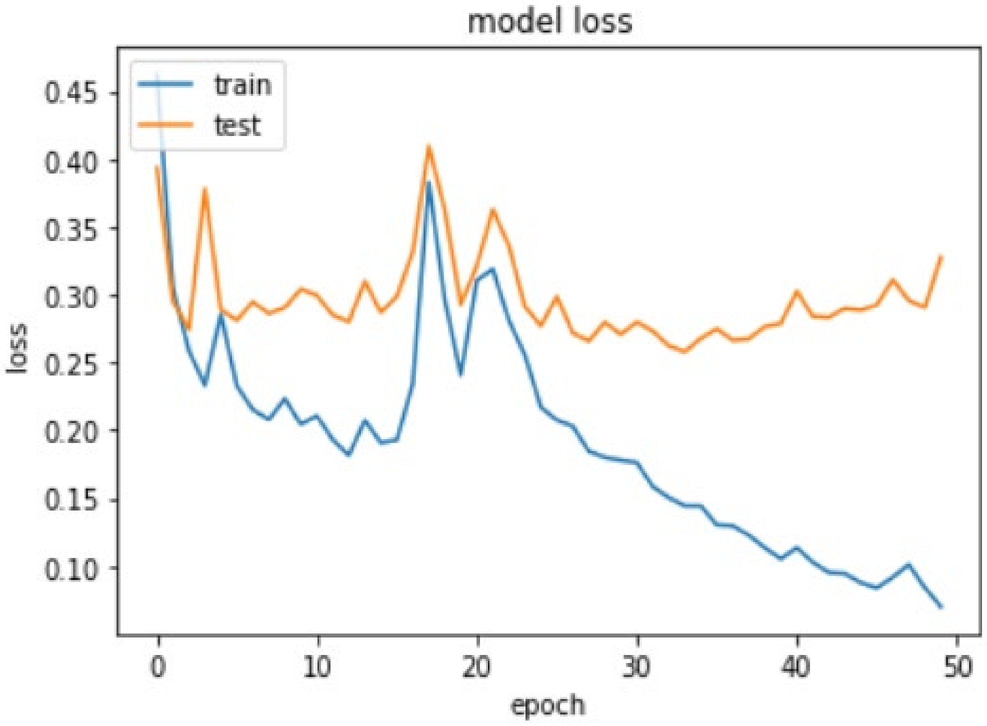
Implementation

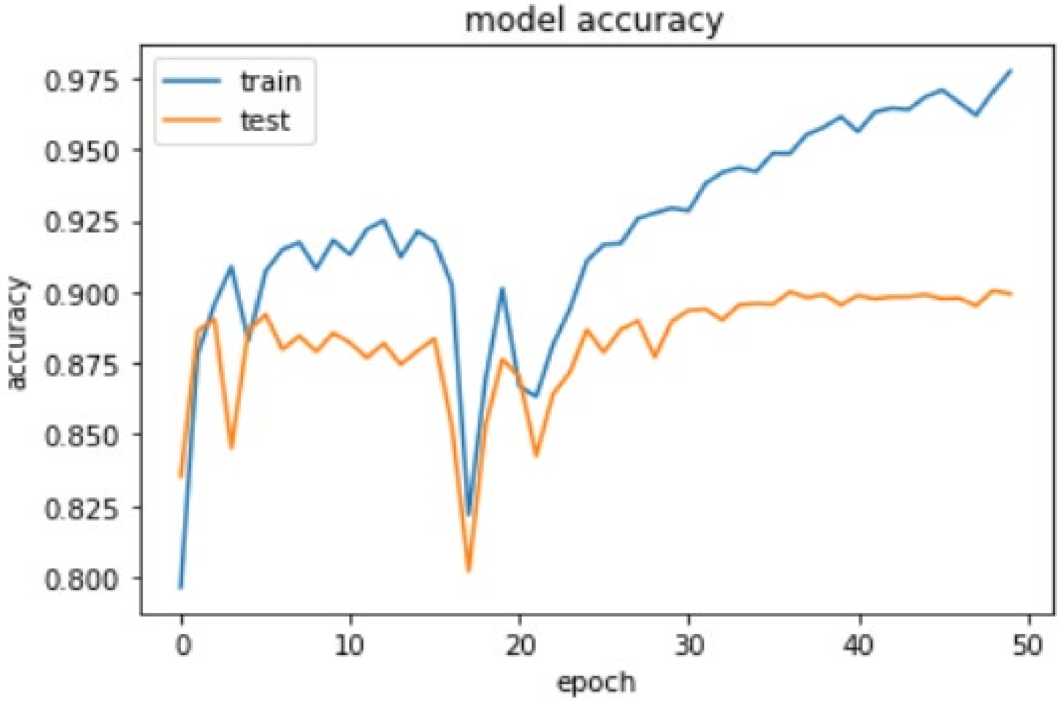
Model-2

- This is an upgrade of previous model with a greater number of features and neurons.
- It has a significant increase in the accuracy but takes a large amount of computation power due to large network.
- It proves our previous deduction that simple LSTM models requires a large number features for getting good results.

Implementation

Model-2





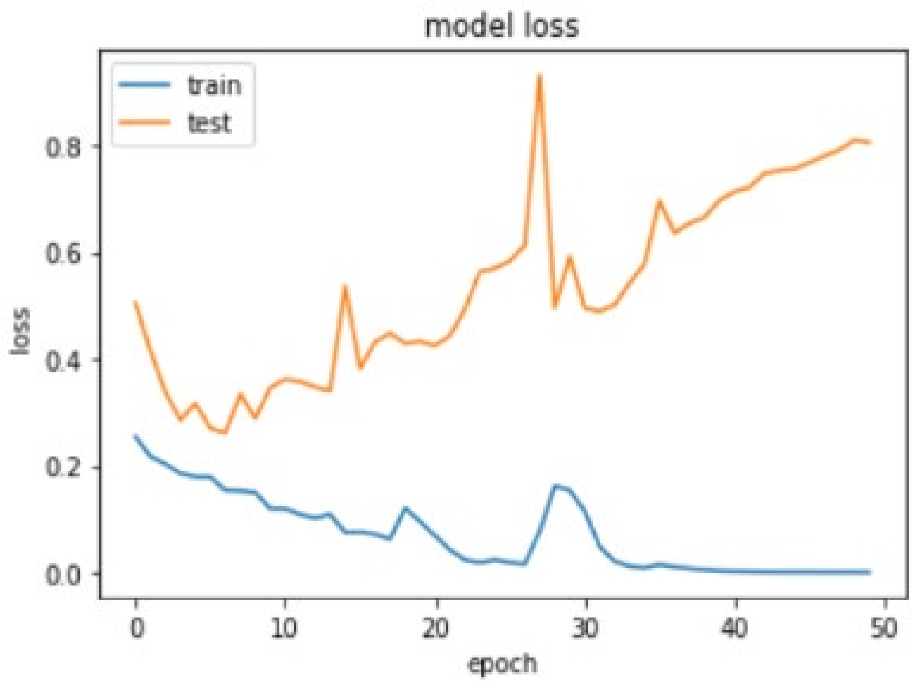
Implementation

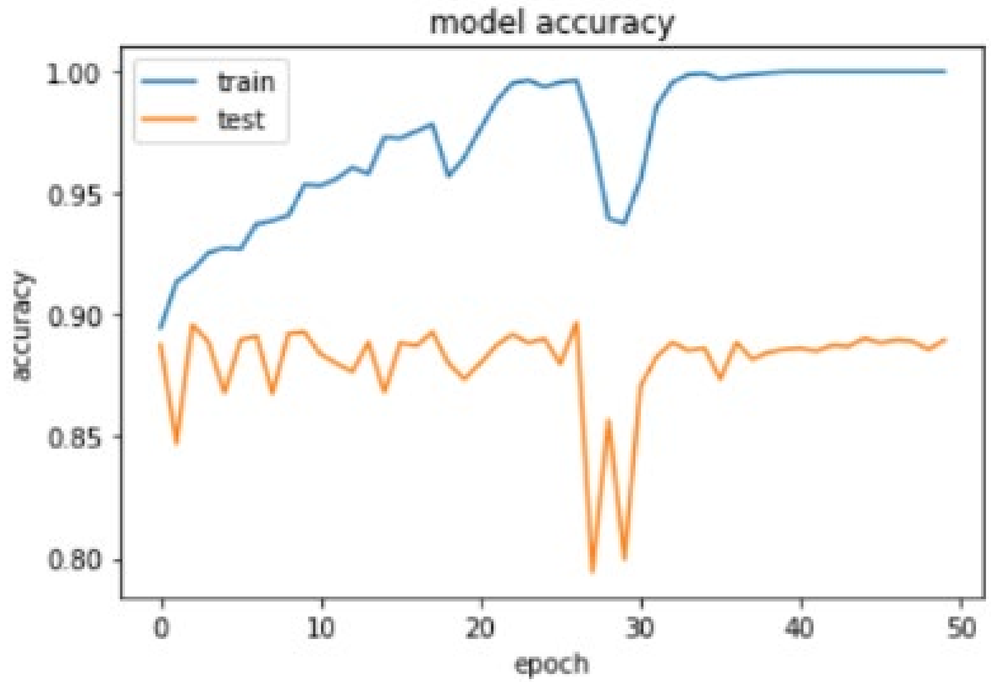
Model-3

- It is an upgrade of previous model with additional batch normalization layer.
- On adding batch normalization, it has boosted the overall accuracy of the previous model.
- It is a technique for training very deep neural networks that normalizes the contributions to a layer for every mini-batch.
- This has the impact of settling the learning process and drastically decreasing the number of training epochs required to train deep neural networks.

Implementation

Model-3





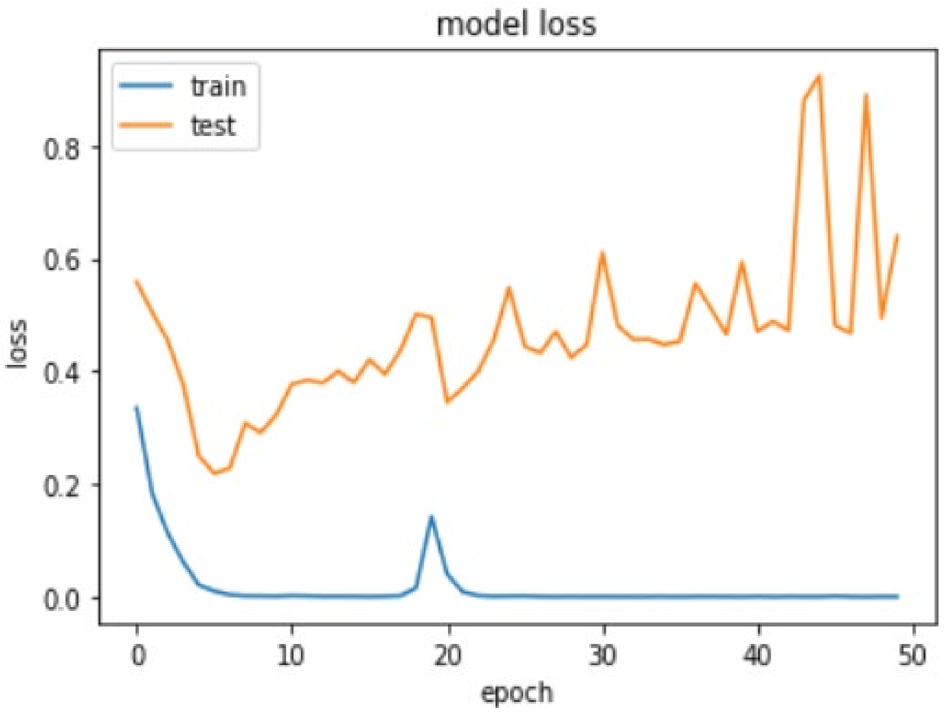
Implementation

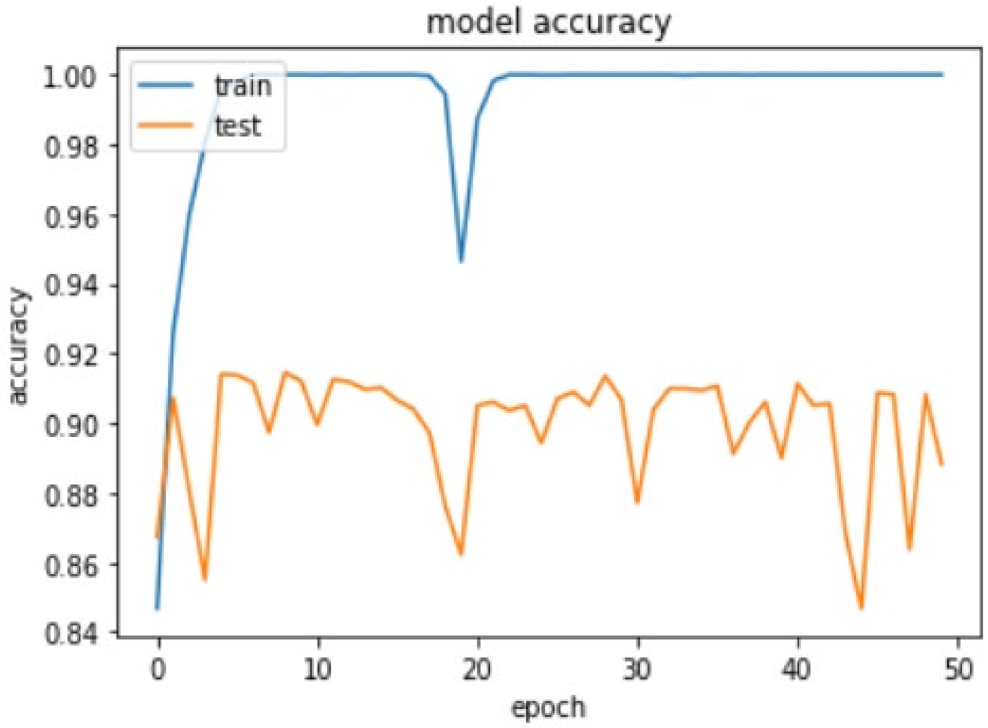
Model-4

- It is an upgrade of previous model with additional CNN network.
- CNN input the is used for detecting the features and then more refined details is passed to LSTM boosting its accuracy to 91%.
- Though we have used batch normalization it is not required that will observe in our final model.

Implementation

Model-4





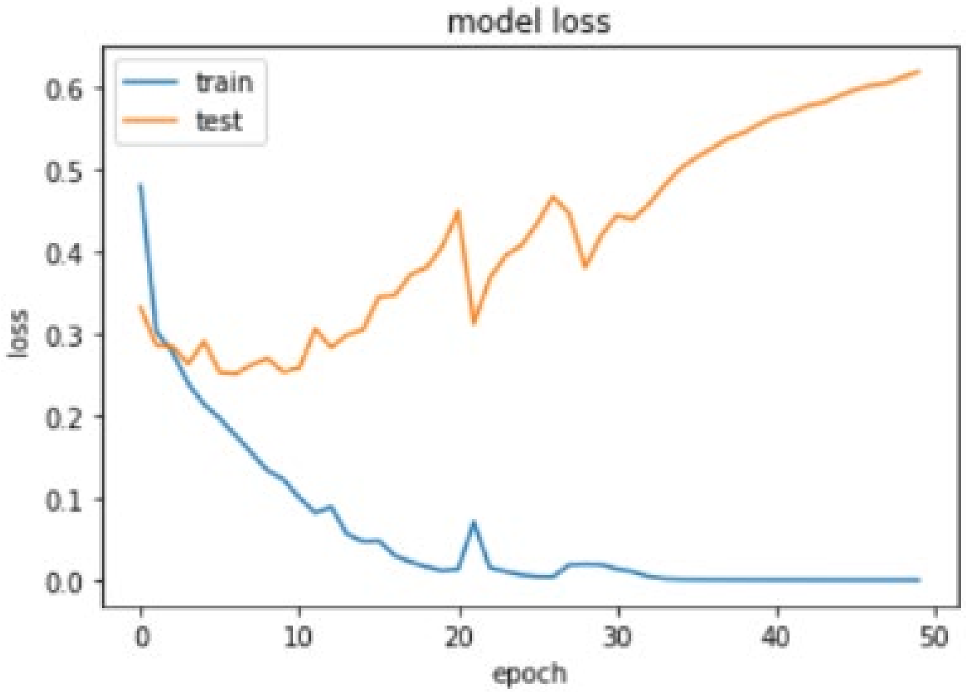
Implementation

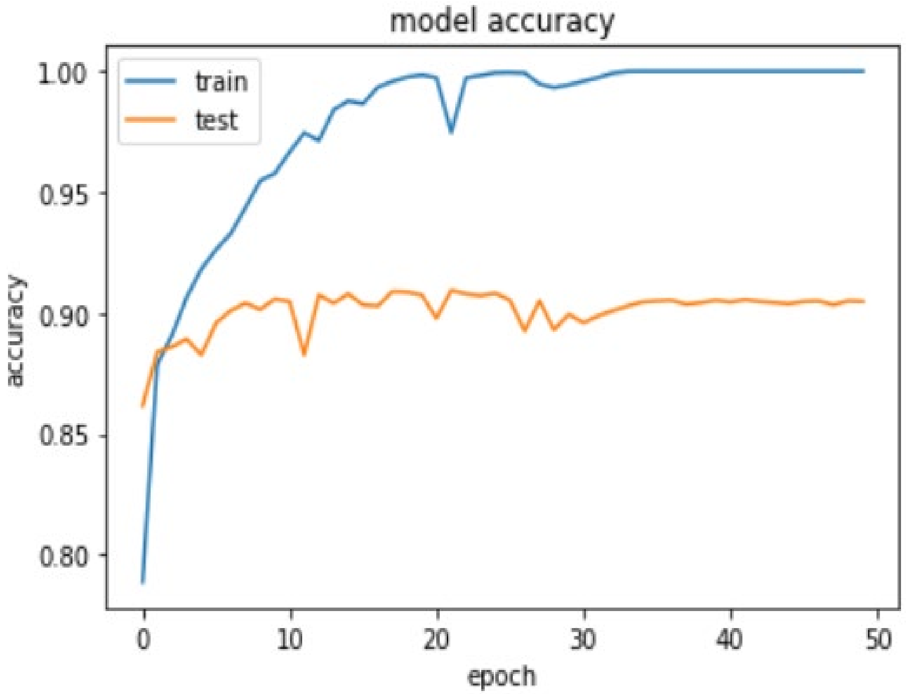
Model-5

- It is a downgrade of previous model with decrease in number of neurons and features and removal of batch normalization.
- We can observe that our final model uses much less computing power and features for same accuracy(as of only LSTM Models).
- Batch normalization is not required as the efficiency remains almost same even on removal of it.

Implementation

Model-5





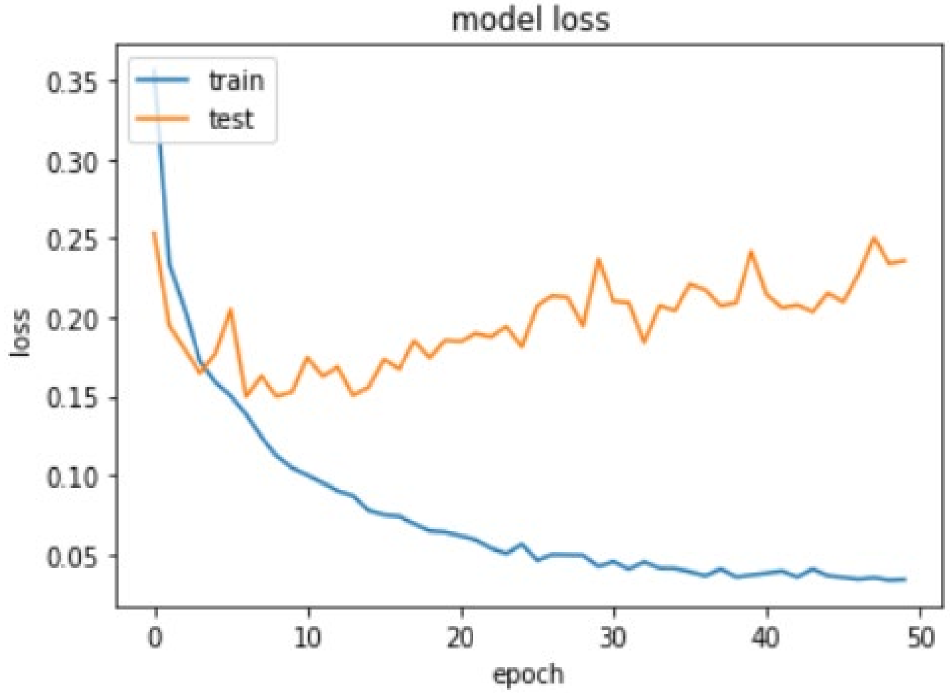
Implementation

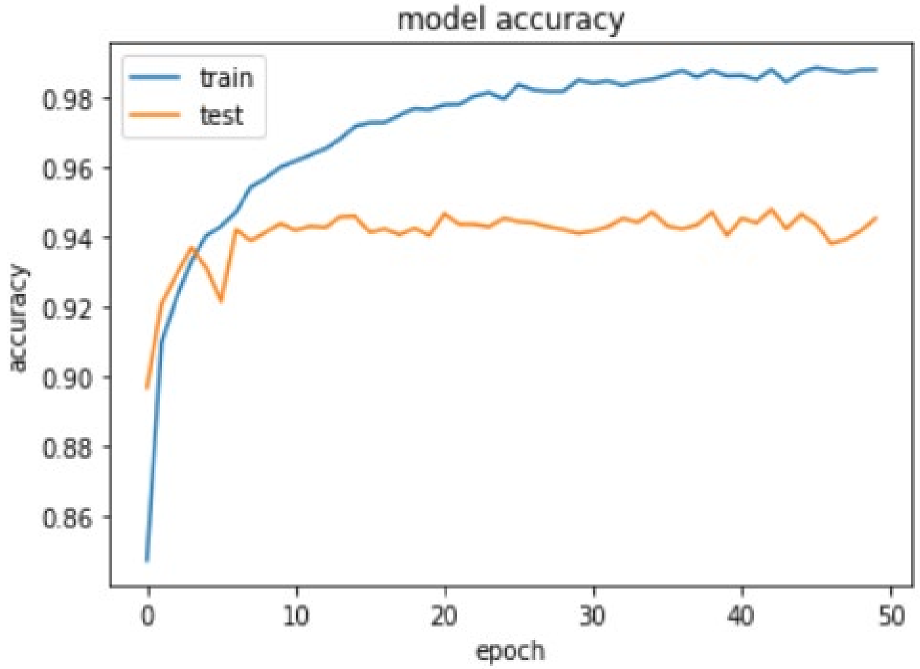
Model-6

- The model with attention layer gets a significant improvement on accuracy.
- The attention layer decides efficiently about the data model needs to focus more on.
- The model is more complex as it uses a greater number of layers than previous ones, but the computational cost for attention layer is less than recurrent layers.

Implementation

Model-6





Implementation

Model-7

- Use of XLM-RoBERTa has boosted the accuracy significantly.
- XLM models specialize in finding hidden sentiments in the texts which led to the best result.
- The model used is a transformer model and takes more RAM and computational cost.

Result and Analysis

Model	Val. Acc.	Train Acc.	Val. Loss	Train Loss	Test Acc.
LSTM only(Model-1) (100 features)	86%	88%	0.30	0.35	84%
LSTM only(Model-2) (300 features)	89%	94%	0.12	0.26	86%
LSTM only(Model-3) (300 features)(Batch Normalization)	89%	98%	0.50	0.10	88%
LSTM + CNN (Model-4) (300 features) (Batch Normalization)	91%	99%	0.50	0.05	90%
LSTM + CNN (Model-5) (100 features)	90%	99%	0.45	0.05	89%
LSTM + Attention (Model-6) (100 features)	94%	99%	0.20	0.04	94%
XLM-RoBERTa (Model-7) (100 features)	98%	99%	0.16	0.02	99%

Conclusion

- In our study we started with simple LSTM models and tested their performance which was proportional to the number of features and the size of network. But still it produced maximum accuracy up to 86%.
- We observed that LSTM + CNN models are capable of producing accuracy up to 89% with much smaller network compared to simple LSTM model.
- We used RNN + Attention model and found that it has accuracy of about 94%.

Conclusion

- While studying attention model further we found that attention model is itself sufficient to give the result.
- Then we switched to transformer models which has accuracy up to 99%. In our study we have able to find out how gradually we progressed from LSTM models to transformer models.
- Our study will help the future researcher to understand the how these models are derived from its predecessor models and what improved its performance from its predecessor.

Future Work

- We are planning to work on detecting fake news shown in form of videos.
- We will be using our knowledge of text-based fake news detection and speech to text conversion.
- We may develop algorithm to identify the fake speaker in the video and then warn the users against him/her.
- We will also work on will contain data pipeline to automate the entire process of fetching data and converting it to required form.

References

[1] Xinyi Zhou and Reza Zafarani. 2020. A Survey of Fake News: Fundamental Theories. https://doi.org/10.1145/3395046

[2] Ayat Abedalla, Aisha AI-Sadi, and Malak Abdullah. 2019. A Closer Look at Fake News Detection. https://doi.org/10.1145/3369114.3369149

[3] Jibran Fawaid, Asiyah Awalina, and Rifky Yunus Krishnabayu . 2021. Indonesia’s Fake News Detection using Transformer Network https://doi.org/10.1145/3479645.3479666

[4] Oluwaseun Ajao, Deepayan Bhowmik, and Shahrzad Zargari. 2018. Fake News Identification on Twitter with Hybrid CNN and RNN Models. https:// doi.org/10.1145/3217804.3217917

[5] S. I. Manzoor, J. Singla and Nikita, "Fake News Detection Using Machine Learning approaches. https://doi.org/10.1109/ICOEI.2019.8862770

References

[6] Zhao, Zhe & Resnick, Paul & Mei, Qiaozhu. (2015). Enquiring Minds: Early Detection of Rumours in social media from Enquiry Posts. 1395-1405. https://doi.org/10.1145/2736277.2741637

[7] Sa, Ahmed & hinkelmann, knut & Corradini, Flavio. (2020). Development of Fake News Model using Machine Learning through Natural Language Processing.

[8] M. Umer, Z. Imtiaz, S. Ullah, A. Mehmood, G. S. Choi and B. -W. On, "Fake News Stance Detection Using Deep Learning Architecture (CNN-LSTM https://doi.org/10.1109/ACCESS.2020.3019735

[9] arXiv:1906.05659 **[cs.CL]**

[10] M. U. Salur and I. Aydin, "A Novel Hybrid Deep Learning Model for Sentiment Classification. https://doi.org/ 10.1109/ACCESS.2020.2982538