**Fake News Article Detection**

**Implementation of LSTM Deep Learning Model**

1. **Main Objective**

Creating finest model to detect fake news article that might lead to hoax or fraud/scam with high accuracy without suffering accuracy paradox.

1. **Data Definition and Description**

This dataset is about news article that classified as fake and legitimate and it is a dataset for competition in Kaggle. Most of the in formation in the dataset is text data

The dataset contain three different file which are training, test, and submit.

The dataset contain 20800 training observations and 5 attributes (variables).

Note : In this research attemp will use only 2000 training observation and 200 test size.

**Data description**

id: unique id for a news article

title: the title of a news article

author: author of the news article

text: the text of the article; could be incomplete

label: a label that marks the article as potentially unreliable

1: unreliable

0: reliable

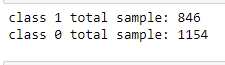
**Research Objective**

Built three model that can predict the legitimation of news articles and select the best one with elaboration of the choosen model.

1. **Pre-processing and Feature Engineering Plan**

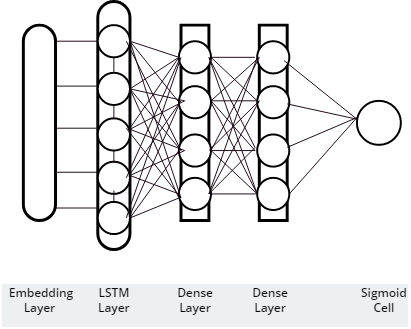
* Data Ingestion from IBM db2 database
* Inspect data and column type
* Inspect null values and fix with dropping them
* Inspect the class balance
* Extract corpus from text column (article content)
* Apply english words stemming
* Apply one hot representation to the corpus
* Apply post padding for LSTM embedding
* Modeling
* Evaluation

**Label classes :**

****

the class balance wont be a huge problem in this research because it relatively in good balance.

**General LSTM Architecture**

****

1. **Model Selection and Training Results**

To achieve best result that match the business objective, we will conduct three different model settings and parameters but the general architecture and deep learning model used remain the same which is Long-Short Term Memory (LSTM)

**Modeling**

1. **LSTM version 1**

**Architecture :**

Vocabulary = 10.000

Sentence length = 48

Embedded Feature representation = 45

Optimizer = Adam

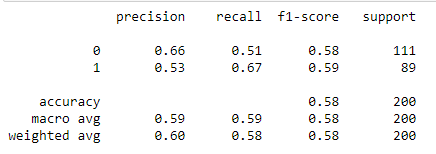
Epoch =5

Batch = 100

Learning rate = 0.03

|  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- |
| Property | Layer | | | | |
| Layer 1 | Layer 2 | Layer 3 | Layer 4 | Layer 5 |
| Type | Embedding/Input | LSTM layer | Dense | Dense | Classifier |
| Activation | - | ReLU | ReLU | ReLu | Sigmoid |
| Amount | 10.000 | 100 | 20 | 20 | 1 |
| Parameters to tune | 450.000 | 58400 | 2020 | 420 | 21 |

Result :



1. **LSTM version 2**

**Architecture :**

Vocabulary = 10.000

Sentence length = 48

Embedded Feature representation = 40

Optimizer = Adam

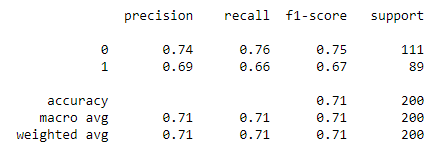
Epoch =5

Batch = 200

Learning rate = 0.03

|  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- |
| Property | Layer | | | | |
| Layer 1 | Layer 2 | Layer 3 | Layer 4 | Layer 5 |
| Type | Embedding/Input | LSTM layer | Dense | Dense | Classifier |
| Activation | - | TanH | ReLU | ReLu | Sigmoid |
| Amount | 10.000 | 150 | 20 | 20 | 1 |
| Parameters to tune | 400.000 | 114600 | 3020 | 420 | 21 |

Result :



1. **LSTM version 2**

**Architecture :**

Vocabulary = 10.000

Sentence length = 48

Embedded Feature representation = 50

Optimizer = Adam

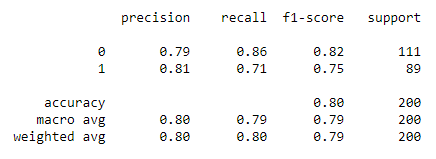
Epoch = 8

Batch = 400

Learning rate = 0.003

|  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- |
| Property | Layer | | | | |
| Layer 1 | Layer 2 | Layer 3 | Layer 4 | Layer 5 |
| Type | Embedding/Input | LSTM layer | Dense | Dense | Classifier |
| Activation | - | TanH | ReLU | ReLu | Sigmoid |
| Amount | 10.000 | 50 | 20 | 20 | 1 |
| Parameters to tune | 400.000 | 18.200 | 1020 | 420 | 21 |

Result



1. **Model Selection, Conclusion, and Summary**

Based on the model version and attempt, we would suggest to use LSTM model version 3 and choose it as the best model in this research with the result ;

* Weighted accuracy = 79 %
* Countable on predicting class 0

From now on, this model will be mentioned as choosen model.

Due to best accuracy main objective, the model training will focus on how to predict a label with high accuracy without suffering accuracy paradox or significant false positif/negative. From the chosen deep learning model **we can conclude below insights that drive accuracy performace**:

1. Learning rate with 3-e4 proven increasing accuracy without suffering false hypothesis
2. Increasing amout of epoch from 5 to 8 will increase accuracy
3. 50 LSTM cell still resulting great accuracy yet faster training

**Further Development**

1. Dataset flaw

* More than thousand important information missing
* Unknown character probably emoticon might cause stemming process fail
* Heterotopic of article content might cause long training
* Slight class imbalance, probably wont significantly affect model accuracy

1. Futher possible research

With dataset open-sourcity and broader techniques, any researcher can expand this report or conduct different research purposes, exploratory data analysis, and many more. Highly encouraged to develop further natural language processing technique on this dataset to generate more accurate prediction.

* Need further experiment on deep learning model used like RNN, LSTM, or probably any model that suit this dataset
* Need further experiment on ‘best’ epoch number
* Need further experiment on another activation dunction
* Need more development on deep learning architecture

**References**

Author’s notebook and source code

<https://dataplatform.cloud.ibm.com/analytics/notebooks/v2/684a02dd-7486-4673-b2f3-abe3f1575411/view?access_token=01240bd4d53e38b62816b4d26e0f1f415e88489afb6701d671fa9e4a76937dbf>

datasets

<https://www.kaggle.com/c/fake-news>

krish naik similiar research

https://github.com/krishnaik06