A Presentation on

”Comparative Performance of Machine Learning Algorithms for Fake News Detection”

Authors: Arvinder Pal Singh Bali, Mexson Fernandes, Sourabh Choubey, and Mahima Goel

Affiliations: APIIT, Haryana, India

Published by: Springer Nature Singapore Pte. Ltd. 2019

Presented by:

1. Md. Abdul Motaleb (St. ID: 1935 369 650)
2. Kawsin Jannet (St. ID: 1935 117 050)
3. Sabrina Yeasmin (St. ID: 1935 074 650)
4. Mohammad Minhazul Alam (St. ID: 1935 365 650)

Automated Fake News Detection – Overview

- Goal: Automatic detection of fake news
- Types of ‘fake news’:
  1. False news
  2. Fake satire news
  3. Poorly written news
- Importance:
  1. Rapid increase of social media use resulted in exponential growth of fake news sources, opinion spams and low quality content
  2. Control is needed for the stability and progress of society

Overview of the Work Done

- Three standard datasets explored -
  - OpenSources, Kaggle, and George McIntire
- A total of 163 features extracted
- Seven machine learning algorithms compared -
  - Random Forest (RF), Support Vector Classifier (SVC), Gaussian Naïve Bayes (GNB), AdaBoost (AB), K-Nearest Neighbor (KNN), Multi-Layer Perceptron (MLP), and Gradient Boosting (XGB)
- Results Analyzed -
  - Gradient Boosting outperformed others with Mean Accuracy of 88% and F1-score of 0.91

Data Sets: Overview

1. OpenSources dataset
    - Total nine million+ articles
2. Kaggle dataset (https://www.kaggle.com/jruvika/fake-news-detection)
3. George McIntire dataset
4. Description below:

Table 1. Labeled dataset description

Dataset	Fake count	Reliable count	Total
Open sources [16]	5385	5776	11161
Kaggle dataset [17]	10413	10387	20800
George McIntire dataset [18]	3164	3171	6335

Data Features used

- N-grams Count Features
  - Occurrences of n-grams in title and body of an article are extracted
  - Ratios like - total unique n-gram / total n-gram are calculated
  - Useful to detect articles not very well written
- Term Frequency – Inverse Document Frequency (tf-idf)
  - Detects key-words of a document
  - Used to detect the key words from document title and body
  - Cosine similarity is used to check the correlation between title and body of each document

Data Features used (contd.)

- Word Embedding
  - Assigns a real-valued vector to each word of the corpus
  - Calculated from aggregated word-word co-occurrence count
  - Represented 400k words in a 50 dimensional vector space
- Sentiment Polarity Score
  - Open source Natural Language Tool Kit (NLTK) used
  - Positive, Negative, Neutral and Compound sentiments
- Readability standard
  - An approximation of years of education required to understand a sentence on single reading

Data Features: Summary

Table 2. Features and their counts.

Features		Number of feature vectors
Sentiment	Headline	4
	Content	4
Readability		12
Count		41
Cosine Similarity of Normalized *tf-idf* vectors between headline and content		1
Word Embedding	Headline	50
	Content	50
	Cosine similarity between Headline and Content	1
Total number of features		163

Data Set: Preprocessing

- Symbolic characters, date, numbers replaced
- URLs removed
- Stop words (am, is, by, the, etc.) removed
- Texts tokenized and Words stemmed
- Unigrams, bigrams, and trigrams created

Classification Model

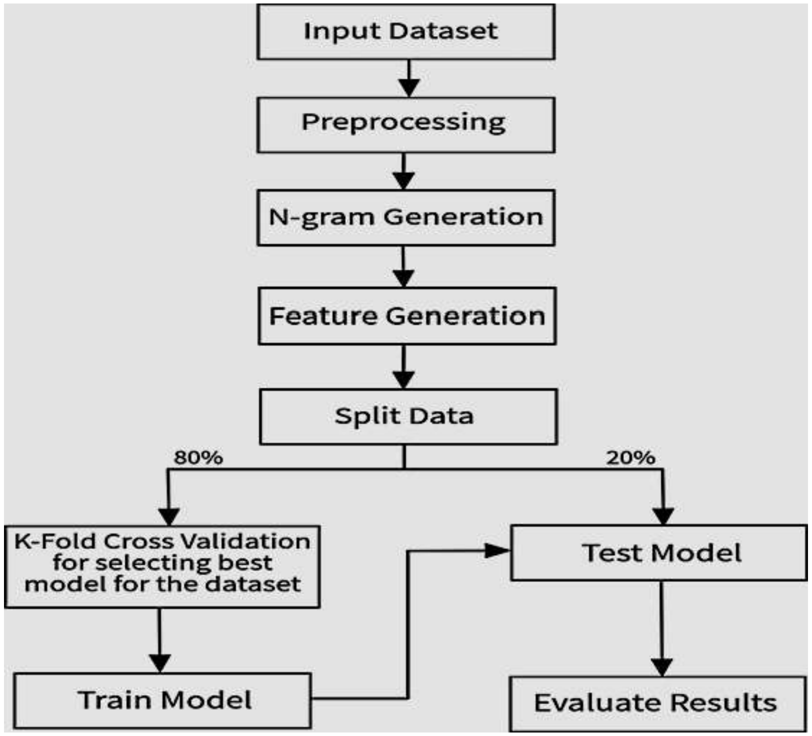
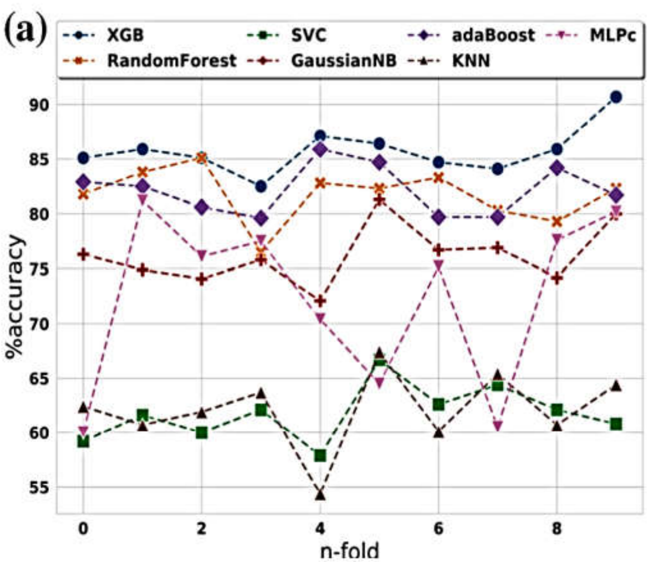


Figure 1: The full classification model

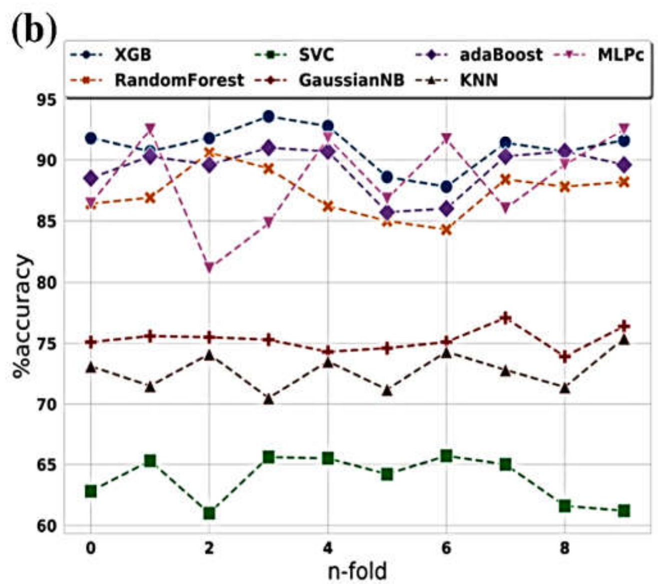
Model Evaluation

- Evaluation Criteria
  - Dataset divided into training and test sets (80-20)
  - Stratified 10-fold cross validation used for selecting the best model from training sets
  - The best model for each algorithm is applied on the test set
- Accuracy and Standard Deviation
  - Accuracy of the best model for the K-folds are computed
  - Standard Deviation of the Accuracies are calculated

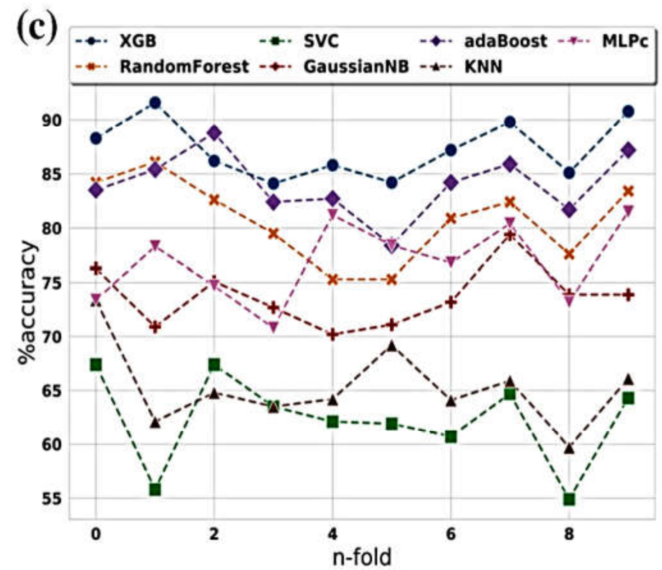
Model Evaluation (contd.)



Data set 1



Data set 2



Data set 3

Figure 2: Model Evaluation

Model Evaluation (contd.)

Table 3: Model Accuracy and Standard Deviation

Classifier	Dataset 1		Dataset 2		Dataset 3	
	A*	SD**	A*	SD**	A*	SD**
XGB	**86.2**	**2.21**	**91.05**	1.67	**87.3**	**2.59**
RF	81.2	2.32	86.63	1.8	82.6	3.1
SVC	62.9	2.31	63.55	1.65	62	3.96
GNB	76.4	2.67	75.24	1.02	73.2	2.73
AB	81.7	2.25	89.25	1.87	83.7	2.75
KNN	62.5	3.28	72.54	1.43	64.8	3.6
MLP	67.1	6.18	88.36	9.93	72.8	6.14

*A: Average Accuracy
**SD: Standard Deviation
Note: All values are in percentage

Performance Measures: Gradient Boosting

- Gradient Boosting (XGB) was found to be the best classification technique based on Accuracy and Standard Deviation
- Precision, Recall and F1-score of Gradient Boosting on the tested datasets are as below -

Table 4: Performance measures of Gradient Boosting

Dataset	Precision	Recall	F1-score
Dataset 1	0.92	0.92	0.92
Dataset 2	0.93	0.94	0.94
Dataset 3	0.89	0.87	0.89

Data Features: Analysis based on output

Table 5: Relevance of selected features

Feature excluded		Dataset 1	Dataset 2	Dataset 3
tf-idf	F1-score	0.9	0.94	0.89
	Loss	0.02	0	0
Count	F1-score	0.89	0.92	0.88
	Loss	0.03	0.02	0.01
Word embedding	F1-score	0.8	0.9	0.85
	Loss	0.12	0.04	0.04
Sentiment	F1-score	0.9	0.94	0.89
	Loss	0.02	0	0
Readability	F1-score	0.9	0.93	0.88
	Loss	0.02	0.01	0.01