News authenticity Analyzer using vectorization

Mayurkumar Patel

University of Wisconsin-Milwaukee

Master's Capstone Project

Dr Jerald R Thomas Jr.

Index

1)Abstract

The fast spread of information in the digital age is like a double-edged sword: it makes knowledge easier to find but also makes it easier for false information to spread. This project presents the "News Authenticity Analyzer using Vectorization," a device meant to stop the spread of fake news. This system, looks closely at news stories using advanced data analytics and machine learning methods. The tool checks the accuracy of information by looking at the text, the reliability of the source, and how the audience interacts with it. It then gives the information a confidence number that shows how true it is. The goal is to give people the tools they need to tell the difference between real and fake material, which will make the public smarter and protect political processes. This essay talks about how the prototype was made, the problems that came up, and the tests that were done on it. It shows how useful it can be in this age of too much information.

2)Introduction

There is more digital media than ever before, which has changed how information is shared and used around the world. Now, news gets to people faster than ever before. But this fast spread of information also comes with big risks, especially the spread of fake information. False information has many effects; it can change people's minds, skew the results of elections, and even start fights. Because of these effects, we need ways to quickly check the accuracy of news stories before they change the way people talk and make decisions.

This project is called "News Authenticity Analyzer using Vectorization," The main goal of the project is to create a high-tech tool that uses the newest data analytics and machine learning methods to check the accuracy of news stories. The system looks at text, viewer engagement data, and source trustworthiness signs to find possible false information and give a confidence level about the news story's accuracy.

3) Objective

The primary aim of the News Authenticity Analyzer project is to develop a robust system that can accurately determine the authenticity of news articles using advanced machine learning techniques. Specific objectives include:

Data Compilation: To gather a comprehensive dataset consisting of verified authentic and fraudulent news stories, providing a solid foundation for model training and testing.

Feature Selection: To employ advanced feature selection techniques to identify key textual elements and source characteristics that are indicative of news authenticity. This will involve exploring various vectorization methods to transform raw text into a format suitable for machine learning analysis.

Algorithm Development: To design and refine a classification algorithm capable of distinguishing between authentic and fake news with high accuracy. This includes the iterative process of training, validating, and tuning the model to enhance its predictive performance.

Model Testing and Evaluation: To conduct thorough testing of the model using both the collected dataset and real-world scenarios to assess its reliability and accuracy. This involves adjusting the model based on feedback and performance metrics to ensure it meets the required standards of precision and recall.

Community Impact: To evaluate how effectively the tool empowers users to identify misinformation, with the goal of enhancing public knowledge and fostering critical thinking about media consumption.

4) dataset

Any machine learning model's performance is mostly dependent on how good and suitable the dataset is for testing and training. We have acquired a dataset from Kaggle for the News Authenticity Analyzer project, which is intended especially for the identification of false news. Accessible to the public at Kaggle: Fake News Detection, this dataset is named "Fake News Detection". A binary categorization system cannot be developed without a well-balanced combination of authentic and fraudulent news pieces. The hundreds of news items in the collection are classified as either "Real" or "Fake." Important characteristics of the dataset that will be used in this work consist on:

Title: The headline of the news article.

Text: The full textual content of the news article.

Label: A binary indicator, where 'Real' is denoted by 1 and 'Fake' by 0, categorizing the authenticity of the article.

Link: https://www.kaggle.com/datasets/bhavikjikadara/fake-news-detection/

5) Preprocessing

achine learning models need effective preprocessing to be able to convert unstructured text into an easily examined format. The advanced methods used with many Python modules in the News Authenticity Analyzer's preparation pipeline are described in this section.

5.1 Stemming

An NLTK library PorterStemmer is used for stemming. By reducing words to their base form, the corpus becomes less complicated and processing efficiency rises. To the root "run," for example, are reduced the words "running," "runner," and "ran." Depending on the linguistic quirks of the dataset, PorterStemmer allows parameters like the mode of operation to be changed to balance aggressive and more cautious stemming.

5.2 Lemmatizing

Sophisticated than stemming, lemmatizing makes use of NLTK's WordNet Lemmatizer. The base or dictionary form of a word is returned by this method taking morphological analysis into account. For instance, all of 'is,' 'am,' and 'are' lemmatize to 'be.' Usually, lemmatizing calls for the specification of each word's parts of speech, which improves the process's accuracy by guaranteeing that the words are understood as they are used in the text.

5.3 Sentiment Analysis

TextBlob is used by the project for sentiment analysis, which works especially well at expressing the subjectivity and polarity of the text. TextBlob gives the text polarity scores (from -1 to 1) and subjectivity scores (from 0 to 1) using a pretrained model. As it helps detect perhaps biassed or emotionally charged language—a signature of deceptive or false news material—this stage is essential to the analysis. To raise the precision of identifying false news, the classification models may use the results of this research as a feature.

5.4 Doc2Vec

Gensim's Doc2Vec transforms texts into vector representations that maintain word semantics in context. The size of the vectors, the window for context words, and the method (either Distributed Memory or Distributed Bag of Words) are among the various adjustable elements of this approach. One may find the best representations for telling true news from fraudulent news by testing with various settings. Doc2Vec produces dense and informative vectors that greatly improve the performance of later classification models.

6. Classification Models

The News Authenticity Analyzer project classified news items as true or false with accuracy using several machine learning models. The theoretical foundations, implementation specifics, and application in this project of each model are presented in depth in this section.

6.1. Basics of Model Building

Understanding how categorization models are typically built and trained is essential before delving into any one model. Generally, the process includes characterizing, choosing a model, training it on a dataset, and then evaluating accuracy, precision, and recall of the model.

6.1.1. Naive Bayes

Additionally naive bayes classifiers, well-known for their effectiveness and efficiency in text categorization. Although simplified, the Bayes' Theorem used in these probabilistic models shows remarkable performance in real-world applications. Their skill in handling data with several dimensions makes them particularly

6.2. Support vector machine.

One strong technique used in this work is Support Vector Machines (SVM). The way Support Vector Machines (SVMs) work is by finding the best hyperplane that efficiently separates the data into numerous groups. Through kernel approaches, which involve converting the input into higher dimensions, SVMs can handle non-linear data. For this project, the implementation of SVM necessitated the optimization of parameters including the regularization parameter C and the choice of kernel type (linear, polynomial, RBF).

6.3. Decision Tree

6.3.1. Decision tree classifier

With its well-known, understandable architecture, decision trees were used to make successive decisions based on the characteristics of the news items. Inside a decision tree, each node symbolizes a specific quality, each branch a decision rule, and each leaf a conclusion. Decision trees may, however, overfit, a problem that has been lessened by techniques like pruning.

6.3.2. Gradient Boosting

This work uses the potent ensemble learning method gradient boosting to improve the resilience and prediction accuracy of our model. More precisely, the speed and model performance optimization of the XGBoost Gradient Boosting implementation was selected.XGBoost additionally supports several regularization methods and has integrated procedures for addressing missing data, which lessen overfitting. Because regularization helps to preserve a simpler model and penalizes big coefficients, it is essential in complicated models.

6.4. Nearest neighbors

By comparing news articles to nearby data points, the K-Nearest Neighbors (KNN) method was utilized to classify them. To find the best configuration, tests were carried out using different values of k and different distance metrics, like Manhattan and Euclidean. Again, the importance of logistic regression in binary categorization was underlined. It converts anticipated values to probabilities by use of the sigmoid function. Less overfitting is achieved by regularization techniques like L1 (Lasso) and L2 (Ridge).

6.5. Regression model

6.5.1. Logistic regression

On the training set, the logistic regression model is trained with parameters that guarantee repeatability by allowing a maximum of 1000 iterations and a fixed random state. Forecasts then are produced using the testing dataset. Three metrics—accuracy, Cohen's Kappa score, and Matthews Correlation coefficient (MCC)—are used to evaluate the model's performance in each folding. These parameters provide a detailed evaluation of the reliability and capabilities of the model to produce precise forecasts. The metrics provide a general evaluation of the model's performance in identifying the target variable across different subsets of the data as they are computed by averaging across all folds.

6.6. Advanced Technique

6.6.1. Ensemble models

Ensemble models, which are advanced machine learning algorithms that use several individual models, may enhance overall performance, robustness, and accuracy of predictions. These methods are particularly valuable in complex scenarios, such as identifying fake news, where the limitations of individual models may be overcome by harnessing the capabilities of several models.

6.6.1.1. Voting based model

Each classifier in the ensemble, namely Logistic Regression (log_clf), Decision Tree (tree_clf), and KNN (knn_clf), is started with a random_state value of 42, where appropriate, to ensure that the findings can be reproduced. The Logistic Regression model offers a reliable foundation with high efficiency for linear issues. The Decision Tree Classifier provides profound insights as a result of its hierarchical structure, which is especially advantageous for managing non-linear connections within the data. The KNN classifier plays a role by collecting local patterns in the dataset, providing a non-parametric method for categorization.

The VotingClassifier is setup with three models and is set to utilize 'soft' voting. During 'soft' voting, the various classifiers' estimated probabilities for each class are summed, and the final prediction is made based on the highest probability. This method exploits the probabilistic characteristics of Logistic Regression and the distance-based certainty of KNN, together with the heuristic class distinctions offered by Decision Trees. This leads to a more adaptable and often more precise categorization method in contrast to 'hard' voting, which simply tallies votes based on labels.

6.6.1.2. Stacking model

With the stacking ensemble methodology, an advanced machine learning method, a final model (second-level or meta-model) is trained on the outputs of the base models to generate a final prediction after many base models (sometimes referred to as first-level models) have been trained to tackle the same issue. The advantages of many models are combined in this approach to get superior prediction performance than any one model working alone.
I choose a GradientBoostingClassifier as the foundation model. The efficiency with which this classifier manages outliers and non-linear features among other data anomalies is well-known. Its configuration of 50 estimators, 0.1 learning rate, and 3 maximum depth strikes a compromise between model complexity and performance. For repeatability, the random state is set at 42.Logistic Regression serves as the last estimator in the StackingClassifier, which has as its foundation the specified

GradientBoostingClassifier. The second level of prediction is made using logistic regression using the GradientBoostingClassifier's output. During training, the stacking model employs fivefold cross-validation (determined by the cv parameter), which guarantees that every dataset instance is utilized for both training and validation and helps avoid overfitting.

6.6.2. Neural networks

Using TensorFlow and Keras, this sequentially constructed neural network is intended for binary categorization. The 256-neuron input layer of the model is enabled by non-sparse gradients using LeakyReLU activation with an alpha of 0.01. After every activation, layer outputs are normalized using batch normalization to stabilize learning. Dropout layers with rates falling from 0.5 to 0.3 are inserted after each batch normalization to prevent overfitting and lessen the possibility of reliance on any one neuron.

The network's complexity decreases with each hidden layer—128, 64, and 32 neurons—keeping the same activation, normalization, and dropout configuration. A sigmoid activation function on a single neuron in the last layer generates a likelihood that inputs will be classified into one of two groups.

Using a learning rate of 0.0001, the Adam optimizer—known for its flexible learning rate capabilities—optimizes the binary crossentropy loss function. Early stopping is used during training to reduce overfitting by pausing after seven epochs if the validation loss does not decrease. Performance and generalization are well balanced in this simplified architecture on

7. Metrics

Throughout the project, several measures have been used to assess the neural network and machine learning models' performance.

1. *Accuracy:* Computed as the percentage of accurate predictions (including true positives and true negatives) over the entire number of instances studied, this is the simplest simple statistic. It offers a fundamental gauge of the general efficacy of a model.
2. *Cohen's Kappa Score*: Adjusted for any chance agreement, this statistic gauges the agreement between two raters (or, in this instance, the projections and the actual numbers). In skewed datasets in particular, it is a more reliable substitute for accuracy.
3. *Matthews Correlation Coefficient (MCC):* MCC is a well-balanced statistic that works even in cases where the classes are somewhat dissimilar in size. It gives back a number between -1 and +1, where -1 denotes complete discrepancy between forecast and observation, 0 no better than random prediction.
4. *Binary Crossentropy:* Measured as a loss function for binary classification models, binary crossentropy quantifies the difference between the probability distribution of the outcomes and the predictions.
5. *Classification Report:* This extensive measure comprises a number of performance indicators:
    a. Precision: The proportion of accurately anticipated positive observations to all anticipated positives. The correctness of the optimistic forecasts is shown.

b. Recall (Sensitivity): The proportion of all real positive observations to those that were accurately projected positive. It shows how successfully the model can locate every occurrence of positivity.
c. F1-Score: Recall and Precision averaged together. Especially helpful when the class distribution is unequal, this score considers both false positives and false negatives.

6. *Validation Loss:* Computed on a separate validation dataset that is not utilized for training, this loss is monitored throughout neural network training. It facilitates model adjustment without excessive fit to the training data
7. *Early Stopping*: Although not strictly speaking a metric, early stopping is a technique used in model training to halt training as soon as the performance on a held-out validation set begins to decline, usually indicated by a rise in validation loss across epochs.

8. Results

Model	**Accuracy**	**Cohen'skappa**	**Matthews correlation**
Naïve bayes	0.9713744709289374	0.9427204202037358	
SVM	0.9858536355612681	0.9716412703685527	0.9716412703685527
Decision tree	0.7581534583330243	0.5146418426038768	0.514831807937079
KNearest Neighbour	0.8604303048857339	0.718605474228497	0.7244780273950833
Logistic regression	0.9890617052519032	0.9780751926899492	0.9780758107446349
Gradient boosting	0.9237435298191441	0.847179932984672	0.8471914729528507
Voting-Based classifier	0.8380485631543774		
Stacking Model	0.8528625529071062		
Neural network	0.9064379334449768		

At about 97.14% accuracy, naïve bayes performed well. Its 0.9427 Cohen's Kappa score likewise indicates a very high degree of agreement over what is predicted by chance.Support Vector Machine (SVM) is successful in exact classifications as shown by its 98.59% accuracy with Cohen's Kappa and Matthews Correlation at 0.9716.Even if decision trees are simple to understand, their comparatively poor performance may be due to overfitting, particularly when working with complicated, high-dimensional datasets like news articles. Because KNN depends on the inherent characteristics of the dataset and requires suitable distance measures and k value selection, it may have had only a modest success in this project.

Strong in binary classification tasks, logistic regression stood out out with an accuracy of 98.91% and Cohen's Kappa and Matthew's Correlation both around 0.978.Along with Cohen's Kappa and Matthews Correlation Coefficient of roughly 0.8472, gradient boosting achieved an accuracy of 92.37%. The solid performance measures of this model show that it gains from the sequential correction of mistakes made by predecessors, which progressively increases prediction accuracy.

An 83.80% accuracy was reported by the Voting Ensemble model, which integrates predictions from many models. While not covered in depth in this review, voting ensembles combine several model outputs to increase prediction stability and lower variation.Eighty-five percent accuracy was obtained via stacking ensemble. This method usually uses the advantages of several underlying models to forecast more precisely, implying a synergistic effect, but the particulars of the base models and meta-learner setups affect its overall effectiveness.

The accuracy of the neural network model, which uses deep learning methods to capture intricate patterns, was 90.64%. Though not the best, this accuracy shows that neural networks—with their layered design and non-linear processing—are fairly successful for text analysis and categorization jobs in noisy datasets like news.

9. Conclusion

Reviewing the performance results of several machine learning models used in the News Authenticity Analyzer project, it is shown that while Logistic Regression and SVM demonstrated remarkable accuracy over 98%, worries regarding possible overfitting need to be taken into account. Though exact, these models may not translate well to fresh, untested data sets. Conversely, the neural network is thought to provide the optimum compromise between high accuracy and generalization capacity, indicating a more reliable model for real-world applications in news authenticity detection.

Furthermore showing the effectiveness of merging many models to improve prediction stability and lower the risk of overfitting were ensemble techniques like Gradient Boosting, Voting, and Stacking. These results emphasize the need of choosing a model that can sustain accuracy in real-world situations in addition to its performance on training data.

10. Future works

The News Authenticity Analyzer project will give top priority in the future to enhance the neural network model's ability to keep high accuracy and better generalize to new data. A complete evaluation of the model's performance will also be ensured by using techniques like cross-validation at every level of model training. Analyzing different activation functions and optimization techniques might help the model become more accurate and efficient. To this end, one may investigate the use of additional layers and dropout techniques to maximize the network's capacity to learn complex patterns without overfitting.

The range of news source article types may also improve the model's learning and prediction capabilities. The textual features of the model may be enhanced by more sophisticated techniques of natural language processing, including entity recognition. An interesting approach to train the algorithm and adjust to changing news writing styles and misinformation is to include real-time data. Moreover, giving the program a straightforward interface will encourage non-technical users to utilize it more.

11. References

- Bharadwaj, A., Ashar, B., Barbhaya, P., Bhatia, R., & Shaikh, Z. (2020). Source based fake news classification using machine learning. Int. J. Innov. Res. Sci. Eng. Technol, 2320- 6710.
- Baria, R., Degadwala, S., Upadhyay, R., & Vyas, D. (2022, February). Theoretical Evaluation of Machine And Deep Learning For Detecting Fake News. In 2022 SecondInternational Conference on Artificial Intelligence and Smart Energy (ICAIS) (pp. 325-329). IEEE
- Dogru, H. B., Tilki, S., Jamil, A., & Hameed, A. A. (2021, April). Deep learning-based classification of news texts using doc2vec model. In 2021 1st International Conference on Artificial Intelligence and Data Analytics (CAIDA) (pp. 91-96). IEEE.
- Ibrahim, Y., Okafor, E., Yahaya, B., Yusuf, S. M., Abubakar, Z. M., & Bagaye, U. Y. (2021, July). Comparative study of ensemble learning techniques for text classification. In 2021 1st International Conference on Multidisciplinary Engineering and Applied Science (ICMEAS) (pp. 1-5). IEEE.
- Rehurek, R., & Sojka, P. (2011). Gensim–python framework for vector space modelling. NLP Centre, Faculty of Informatics, Masaryk University, Brno, Czech Republic, 3(2).
- Mungra, D., Agrawal, A., & Thakkar, A. (2019). A voting-based sentiment classification model. In *Intelligent Communication, Control and Devices: Proceedings of ICICCD 2018* (pp. 551-558). Singapore: Springer Singapore.
- Lai, S., Xu, L., Liu, K., & Zhao, J. (2015, February). Recurrent convolutional neural networks for text classification. In *Proceedings of the AAAI conference on artificial intelligence* (Vol. 29, No. 1).
- Lyu, S., & Liu, J. (2021). Convolutional recurrent neural networks for text classification. *Journal of Database Management (JDM)*, *32*(4), 65-82.
- Xu, H., Dong, M., Zhu, D., Kotov, A., Carcone, A. I., & Naar-King, S. (2016, October). Text classification with topic-based word embedding and convolutional neural networks. In *Proceedings of the 7th ACM International Conference on Bioinformatics, Computational Biology, and Health Informatics* (pp. 88-97).
- Tong, S., & Koller, D. (2001). Support vector machine active learning with applications to text classification. *Journal of machine learning research*, *2*(Nov), 45-66