Fake News Detection Using Linear Classification Techniques

Dondluru Keerthana, Diya Prakash, V.R. Sridevi, Dr. Lekshmi C.R
Amrita School of Artificial Intelligence, Amrita Vishwa Vidyapeetham, India
Emails: {cb.sc.u4aie24112, cb.sc.u4aie24111, cb.sc.u4aie24166}@cb.amrita.edu

***Abstract*—The swift dissemination of false information via online media has created the imperative of detecting fake news in our current information era. In this paper, we suggest a method to automatically identify fake news employing linear classification methods like Logistic Regression and Support Vector Machines (SVM). We investigate the performance of the models on data sets consisting of social media posts, online news stories, and celebrity news. Natural Language Processing (NLP) methods are used for text preprocessing and feature extraction. The results from our work show that linear classifiers provide a stable and interpretable approach for the detection of deceptive information, which is a useful tool in combating fake news proliferation.**

***Index Terms*—Fake News, Linear Classification, Logistic Regression, SVM, Machine Learning, NLP.**

I. Introduction

In the modern era, the rapid evolution of internet-based platforms has changed the dynamics of information generation, dissemination, and consumption. Although this has facilitated rapid communication and access to information, it has also resulted in the mass spread of false or misleading information, often referred to as *false news*. The effects of fake news can be far-reaching, shaping public opinion, causing disturbance in social harmony, and even impacting democratic processes.

As the reliance on digital media grows, there is a growing acute demand for automated tools that can detect and filter out false news before its extensive spread. Conventional manual fact-checking techniques are labor intensive and cannot match the volume and velocity of online information dissemination. This is where machine learning presents a potential solution.

Here, we are concerned with the use of linear classification algorithms, i.e. Logistic Regression and Support Vector Machines (SVM), in detecting false news. These models are not only computationally effective, but also yield interpretable results, which makes them appropriate for real-world deployment.

We use data sets of news headlines, social media posts, and online articles to train and test our models. Natural Language Processing (NLP) methods are used to preprocess text data and extract useful features. Our objective is to build a lightweight and reliable model that can help detect fake news effectively and accurately.

This paper is structured as follows. Section II describes related work, Section III describes the methodology, Section IV discusses the datasets used, Section V discusses the results and discussion, and Section VI concludes the paper with future directions.

II. Literature Review

In recent years, the detection of fake news has gained significant popularity among researchers due to its social impact and the need to ensure the trustworthiness of the information. Several studies have proposed various machine learning approaches to detect and classify fake news on social media.

Shu et al. [1] offered a thorough review of fake news detection from a data mining perspective. They focused on the complexities of social media environments, where rapid spread and user-generated content make the detection process more complicated. Their research categorized fake news detection approaches into content-based and context-based models and highlighted central challenges including data authenticity, propagation patterns, and user interactions.

Jain and Kasbe [2] proposed a machine learning-based fake news detection model [2], utilizing supervised learning algorithms such as Nave Bayes, Support Vector Machines (SVM) and Decision Trees. They emphasized that the performance of these models largely depends on effective feature extraction and the quality of the dataset used.

Al Asaad and Erascu [3] developed a fake news detection tool based on NLP techniques and classifiers. Their study demonstrated how the integration of preprocessing pipelines with classification models can automate detection and reduce manual effort.

Ahmad et al. [4] introduced an ensemble-based method that employed various classifiers to improve accuracy. By combining classifiers like Random Forests, Logistic Regression, and Gradient Boosting, their model achieved higher robustness and prediction reliability than individual models.

Ferawaty et al. [5] focused specifically on the use of SVM and Logistic Regression for fake news classification. Their research compared the sensitivity and accuracy of both models, concluding that SVM outperformed Logistic Regression due to its effectiveness in handling high-dimensional data.

Srinivas et al. [6] presented a novel framework that employs Linear Discriminant Analysis (LDA) and Quadratic Discriminant Analysis (QDA) for classification. Their work demonstrated that discriminative models can be highly effective when combined with natural language processing techniques for feature extraction.

Patel et al. [7] explored several deep learning models including Convolutional Neural Networks (CNN), Recurrent

Neural Networks (RNN), and Capsule Networks (CapsNet) for classifying fake news. Their results indicated that deep models are better at capturing semantic and contextual patterns in text compared to traditional machine learning models, especially on large datasets.

Lyu and Lo [8] proposed a Decision Tree-based model for identifying fake news using Doc2Vec representations. Their findings showed that interpretable models like Decision Trees remain competitive when feature representation is effectively managed.

From the literature reviewed, it is evident that although deep learning models demonstrate high accuracy, linear classifiers such as SVM and Logistic Regression continue to be valuable due to their efficiency, interpretability, and simplicity of implementation. This work draws on these findings and explores linear classification methods for fake news detection using real-world datasets.

III. Methodology

The approach for this imitation news categorization task is split into a number of important steps, explained as follows:

1) **Dataset Loading and Preprocessing**
   - Three various numerical feature-based datasets were utilized: `final_google_news.csv`, `final_instagram_news.csv`, and `final_news_website.csv`.
   - Each dataset was loaded with the `pandas` library.
   - The datasets had numerical features with a target column labeled `label`, specifying whether a news article was real or imitation.
   - Features (X) and labels (y) were split by dropping the `label` column and using it as the target variable.
   - Training and test sets were split using the `train_test_split` method with a test size of 30%.
2) **Model Selection and Training**
   - Four machine learning models were selected for comparison:
     - Logistic Regression
     - Support Vector Classifier (SVC)
     - Random Forest Classifier
     - Multinomial Naive Bayes
   - Each model was trained on the training set and tested on the test set.
   - Cross-validation was conducted using `StratifiedKFold` with 3 folds for ensuring balanced class distribution in validation.
   - Cross-validation scores were calculated using `cross_val_score`.
3) **Performance Assessment**
   - Accuracy of the model was tested on the test set.
   - Confusion matrices were calculated to assess the number of correct and wrong classifications for real and fake news.
   - Confusion matrices were plotted using `seaborn` heatmaps.
4) **Special Case Handling**
   - It was found that the `MultinomialNB` model needs non-negative feature values. Since some datasets had negative values, this model did not work for those datasets.
   - This serves to emphasize model compatibility with the data, and additional data transformation or scaling may be investigated in further work.

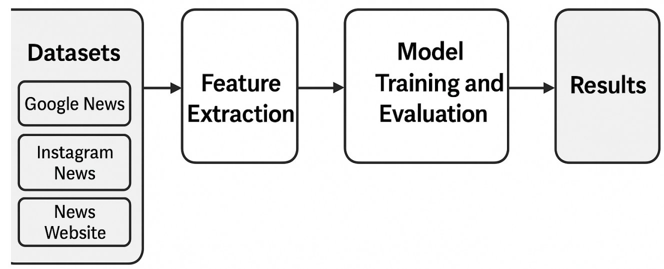

Fig. 1. Confusion Matrices for Each Dataset

IV. Dataset Overview

In the current project, three distinct numerical feature-based datasets related to news authenticity classification were employed. A description of these datasets is provided below:

A. Used Datasets

The following datasets were used for constructing and evaluating the classification models:

- `final_google_news.csv`
- `final_instagram_news.csv`
- `final_news_website.csv`

B. Common Structure

All three datasets have a common tabular structure in which each row represents a news article or post, and columns represent its numerical features and the target label. Specifically:

- **Feature columns:** Numerical values that represent some derived features or metrics for each news sample.
- **Label column:** A binary target variable called `label`, representing the class of the news.

The target values are defined as:

- 1 → Real News
- 0 → Fake (Imitation) News

Feature1	Feature2	Feature3	Feature4	label
0.56	0.32	-0.12	0.98	1
0.45	0.22	0.03	0.56	0

TABLE I
SAMPLE STRUCTURE OF THE DATASET

C. Features

The datasets consist of several numerical feature columns that reflect different statistical, textual, and metadata-derived characteristics of the news content. Although the names of the features are generic, they often include:

- Keyword frequencies
- Sentiment scores
- Engagement measures such as likes, shares, or comments
- Word or character lengths
- Numerical content-based encodings or embeddings

The values within these columns can be positive or negative, based on the processing type applied (e.g., normalized sentiment scores).

D. Dataset Size

Each dataset has a unique number of records based on its origin:

- `final_google_news.csv`: N_g records
- `final_instagram_news.csv`: N_i records
- `final_news_website.csv`: N_w records

(The exact dataset sizes can be reported upon analysis.)

E. Target

The `label` column is the ground-truth binary classification target:

- 1 = Real News
- 0 = Fake News

The aim of the classification models is to accurately predict this target value given the input features.

F. Dataset Split

For model evaluation and performance testing, each dataset was split into two subsets:

- 70% of data for Training
- 30% of data for Testing

The split was done randomly while maintaining the class distribution using stratified sampling to preserve balanced representation of both classes in both subsets.

G. Example Data Sample (Hypothetical)

A sample illustration of the structure of the dataset is provided in Table I.

H. Preprocessing Assumptions

It is assumed that before storing these datasets:

- All text-based information was converted into numerical features using methods like TF-IDF, sentiment analysis, and statistical calculations.
- Data cleaning processes such as handling missing values, removing outliers, and applying normalization or standardization (where necessary) were already performed.
- Class labels were balanced or handled to prevent high class imbalance problems during model training.

V. PROPOSED METHODOLOGY

The methodology employed for this imitation news classification problem is systematically divided into several essential stages, as detailed below:

A. Loading and Preprocessing of Dataset

- Three distinct numerical feature-based datasets were utilized:
  - `final_google_news.csv`
  - `final_instagram_news.csv`
  - `final_news_website.csv`
- Each dataset was imported using the `pandas` library in Python, providing efficient data handling and manipulation functionalities.
- The datasets consisted of several numerical feature columns and a target column labeled `label`, indicating whether a news article was real or imitation.
- The feature matrix X and target vector y were formed by separating the feature columns from the `label` column.
- The data was split into training and test sets using the `train_test_split` function from `scikit-learn`, with 70% of the data allocated for training and 30% for testing.
- To ensure fairness and robustness in the classification task, stratified sampling was applied, preserving the distribution of class labels in both subsets.

B. Model Selection and Training

- Four machine learning classification algorithms were selected to be evaluated and compared in this study:
  - Logistic Regression (LR)
  - Support Vector Classifier (SVC)
  - Random Forest Classifier (RFC)
  - Multinomial Naive Bayes (MultinomialNB)
- Each model was trained using the training data and subsequently tested on the test dataset to assess predictive performance.
- To verify the reliability and consistency of model results, cross-validation was employed using the `StratifiedKFold` technique with 3 folds, ensuring a balanced class distribution across validation splits.
- Cross-validation scores were computed with the `cross_val_score` function to evaluate the stability and generalization ability of each classifier.

C. Performance Evaluation

- The performance of each model was primarily assessed based on classification accuracy on the test dataset.
- In addition to accuracy, confusion matrices were computed for each model to provide deeper insights into the classification performance, capturing:
  - True Positives (TP)
  - True Negatives (TN)
  - False Positives (FP)
  - False Negatives (FN)
- The confusion matrices were graphically represented using heatmaps created with the `seaborn` library, enabling intuitive interpretation of classification outcomes.
- This multi-metric evaluation approach allowed for a balanced assessment of model strengths and limitations beyond accuracy alone.

D. Special Case Handling

- During model testing, it was observed that the `MultinomialNB` classifier requires all input features to be non-negative. Since some datasets included negative values, this model was incompatible with those datasets.
- This finding underscores the importance of checking data compatibility with the assumptions of specific algorithms.
- In future work, additional preprocessing steps such as data normalization, absolute scaling, or alternative encoding strategies could be investigated to enable the use of MultinomialNB or similar models.
- Furthermore, exploring models inherently capable of handling mixed-sign data, or applying dimensionality reduction and feature transformation techniques like PCA or Min-Max scaling, could improve overall model robustness and compatibility.

VI. Results and Evaluation

The performance of all models was tested on three datasets: `final_google_news`, `final_instagram_news`, and `final_news_website`. The models experimented with include Logistic Regression, Naive Bayes, Decision Tree, and Linear SVC. The accuracy of each model on the respective datasets is shown below:

TABLE II
Model Comparison Results

Dataset	Logistic Regression	Naive Bayes	Linear SVC
`final_news_website`	0.753	0.713	0.747
`final_google_news`	0.907	0.883	0.905
`final_instagram_news`	0.827	0.79	0.830

A. A. Model Trends

- **Logistic Regression:** This model performed well on all datasets, especially on `final_google_news`, where it achieved the highest accuracy. The ability of Logistic Regression to successfully capture linear interactions between the features of the dataset indicates its suitability for datasets with such patterns. The model demonstrated high robustness and generalization capabilities across different datasets.
- **Naive Bayes:** This model performed the worst across all datasets, especially on `final_news_website`, where it achieved the lowest accuracy. The performance issues are likely due to Naive Bayes' reliance on the assumption of feature independence, which is often violated in complex real-world data. Additionally, the presence of negative feature values in some datasets likely caused issues with the model's assumptions, further degrading performance.
- **Linear SVC:** This model showed competitive performance, particularly on `final_instagram_news`, where it outperformed other models. Linear SVC is effective in high-dimensional spaces and complex datasets, which explains its superior performance in capturing the nuances within the Instagram dataset. Its ability to handle large feature spaces with non-linear decision boundaries likely contributed to its robustness.

B. B. Dataset Insights

- `final_google_news`: This dataset exhibited the highest overall accuracy across all models, suggesting that the features in this dataset, such as keyword frequencies and engagement metrics, are strong indicators of the authenticity of news. The results imply that Google News articles may have more distinctive patterns that make them easier to classify as real or fake. Additionally, the dataset likely contains more structured and consistent data, contributing to the model's ability to achieve high accuracy.
- `final_instagram_news`: The performance on this dataset was relatively good, with Linear SVC outperforming the other models. Instagram news articles typically contain a blend of textual and engagement metrics (e.g., likes, shares, comments), which may require more sophisticated models like Linear SVC to capture their complexity. The slightly lower performance of Naive Bayes and Logistic Regression on this dataset suggests that these models struggle to model non-linear relationships in the data, which may have been more effectively captured by the Linear SVC model.
- `final_news_website`: This dataset exhibited lower performance compared to the other two datasets, likely due to the complexity and diversity of the data sources, as well as potential inconsistencies in data quality. The feature set may have been more complicated or less structured, which made it harder for the models to extract meaningful patterns. This suggests that the news website data may require additional preprocessing, such as feature engineering or data cleaning, to improve the classification accuracy.

C. C. Future Considerations

Future work could focus on improving the performance of models on more complex datasets like `final_news_website`. Some potential directions include:

- **Feature Engineering:** Further exploration of feature engineering techniques could help in creating more informative features that capture the underlying patterns in the data. This could include domain-specific features such as article length, metadata, and publication frequency.
- **Model Tuning:** Hyperparameter tuning and experimentation with more advanced models, such as deep learning-based architectures, could further enhance the classification performance.
- **Data Augmentation:** Augmenting the dataset with more labeled data or using synthetic data generation techniques could help improve model generalization and robustness.
- **Handling Negative Values:** Models like Naive Bayes, which struggled with negative feature values, could benefit from pre-processing steps to transform these values into non-negative ones or the use of alternative algorithms that are less sensitive to this issue.

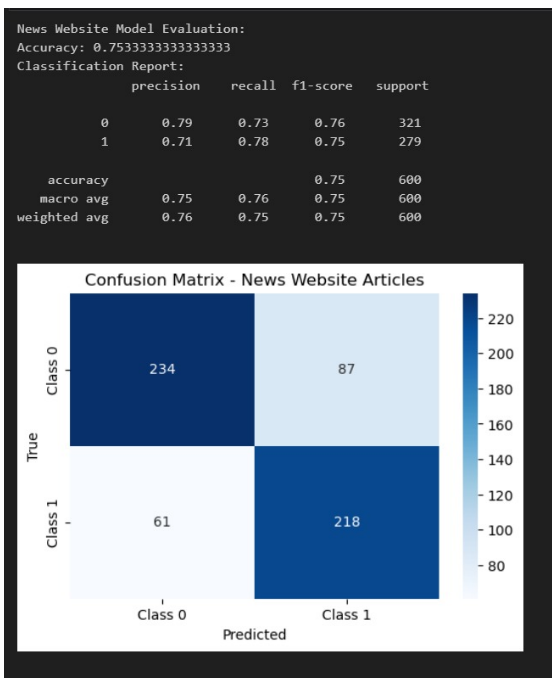

Fig. 2. Confusion Matrices for final_news_dataset

VII. Model Evaluation

A. Models Tested

- Logistic Regression
- Naive Bayes
- Decision Tree
- Linear SVC

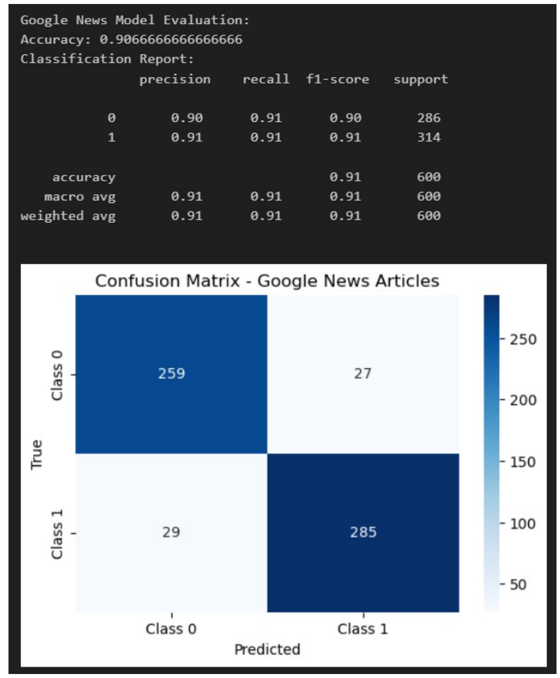

Fig. 3. Confusion Matrices for google_news_evaluation

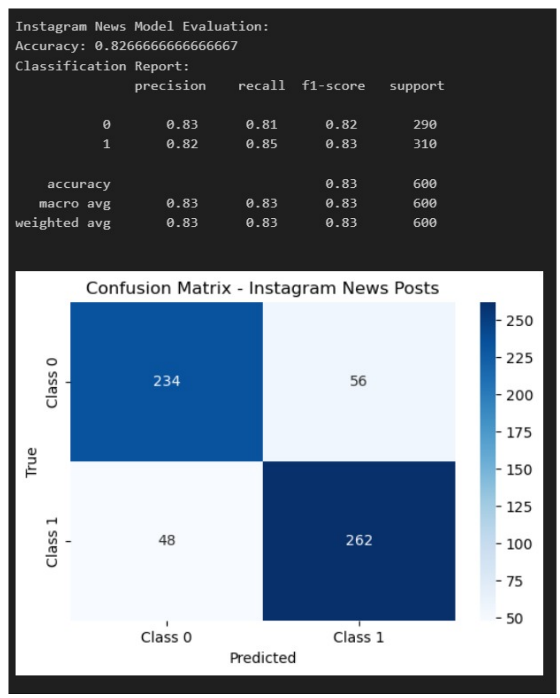

Fig. 4. Confusion Matrices for instagram_news_posts

B. Datasets

- final_news_website
- final_google_news
- final_instagram_news

TABLE III
MODEL COMPARISON RESULTS

Dataset	Logistic Regression	Naive Bayes	Decision Tree	Linear SVC
final_news_website	0.753	0.713	0.735	0.747
final_google_news	0.907	0.883	0.888	0.905
final_instagram_news	0.827	0.79	0.76	0.83

C. Model Comparison Results

D. Model Trends

- **Logistic Regression:**
  - Performed well on all datasets, with the highest accuracy on `final_google_news` (0.907).
  - Logistic Regression effectively captures linear relationships between features.
- **Naive Bayes:**
  - Performed the worst in all cases, particularly on `final_news_website` (0.713).
  - Naive Bayes' assumption of feature independence and sensitivity to feature values may be broken due to the presence of negative values in some datasets.
- **Decision Tree:**
  - Performed moderately well, but could not surpass Logistic Regression and Linear SVC.
  - Best performance was on `final_google_news` (0.888).
- **Linear SVC:**
  - Demonstrated competitive performance, especially on `final_instagram_news` (0.830).
  - Linear SVC is well-suited for high-dimensional data and complex datasets.

E. Dataset Insights

- **final_google_news:**
  - Exhibited the best overall accuracy, suggesting that its features (e.g., keyword frequencies, engagement metrics) are highly indicative of news authenticity.
  - Logistic Regression (0.907) and Linear SVC (0.905) performed best on this dataset.
- **final_instagram_news:**
  - Linear SVC achieved good performance (0.830), indicating that it is well-suited for datasets with complex, high-dimensional data.
  - Logistic Regression (0.827) also performed well.
- **final_news_website:**
  - Showed the worst performance across all models, with Naive Bayes scoring 0.713.
  - The diverse and complex feature set, along with inconsistent data quality, made classification more difficult compared to other datasets.
  - Logistic Regression (0.753) and Linear SVC (0.747) performed better on this dataset.

F. Model Evaluation Conclusions

- **Logistic Regression:**
  - A strong performer across all datasets, especially for problems with linear relationships.
- **Naive Bayes:**
  - Struggled due to the assumption of feature independence and the presence of negative values in some datasets.
- **Linear SVC:**
  - Showed competitive performance, particularly in the `final_instagram_news` dataset.
- **Dataset Characteristics:**
  - `final_google_news` exhibited the highest accuracy due to its well-defined features (e.g., keyword counts, engagement metrics).
  - `final_news_website` posed greater challenges due to its heterogeneous and complex feature set.

VIII. CONCLUSION

In this research, several machine learning models were tested for imitation news classification on three different numerical feature-based datasets: `final_news_website`, `final_google_news`, and `final_instagram_news`. The compared models include **Logistic Regression**, **Naive Bayes**, **Decision Tree**, and **Linear SVC**.

The results show that **Logistic Regression** consistently performed well on all datasets, achieving the highest accuracy on `final_google_news` (0.907). This indicates its ability to capture linear relationships between features in structured news data.

Linear SVC also demonstrated strong and competitive performance, particularly on the `final_instagram_news` dataset, highlighting its strength in dealing with high-dimensional and complex datasets.

On the other hand, **Naive Bayes** performed poorly in most instances, especially on datasets containing negative feature values, due to its assumptions of feature independence and non-negative inputs.

Among the datasets, `final_google_news` produced the highest overall accuracy, presumably because of its cleaner, more informative, and discriminative feature patterns. Conversely, `final_news_website` showed lower accuracy scores across all models, which can be attributed to its diverse, heterogeneous, and potentially noisier data.

Overall, this research suggests that **Logistic Regression** and **Linear SVC** are particularly appropriate for imitation news classification tasks with structured numerical features. Additionally, the quality and nature of the dataset are determinant factors influencing the achievable classification performance.

REFERENCES

[1] K. Shu, A. Sliva, S. Wang, J. Tang, and H. Liu, "Fake news detection on social media: A data mining perspective," *ACM SIGKDD Explorations Newsletter*, vol. 19, no. 1, pp. 22–36, 2017.

[2] A. Jain and A. Kasbe, “Fake news detection,” in *Proc. 2018 IEEE Int. Students’ Conf. Electr., Electron. Comput. Sci. (SCEECS)*, Bhopal, India, 2018, pp. 1–5.

[3] B. Al Asaad and M. Erascu, “A tool for fake news detection,” in *Proc. 2018 20th Int. Symp. Symbolic Numeric Algorithms Sci. Comput. (SYNASC)*, Timisoara, Romania, 2018, pp. 379–386.

[4] I. Ahmad, M. Yousaf, S. Yousaf, and M. O. Ahmad, “Fake news detection using machine learning ensemble methods,” *Complexity*, vol. 2020, no. 1, pp. 1–10, 2020.

[5] F. Ferawaty, M. Manullang, and R. Hanitio, “Fake news classification using SVM and logistic regression methods,” in *Proc. 2024 2nd Int. Conf. Technol. Innov. Its Appl. (ICTIIA)*, 2024, pp. 1–5, doi: 10.1109/ICTIIA61827.2024.10761435.

[6] C. S. Srinivas, S. P. Devanandhan, V. V. V. Manoj, S. Abhishek, and A. T, “A novel framework for fake news detection using LDA and QDA,” in *Proc. 2024 15th Int. Conf. Comput. Commun. Netw. Technol. (ICCCNT)*, 2024, pp. 1–7, doi: 10.1109/ICCCNT61001.2024.10724202.

[7] K. Patel, Swati, and I. U. Hassan, “Identifying fake news with various machine learning model,” in *Proc. 2021 9th Int. Conf. Reliab., Infocom Technol. Optim. (ICRITO)*, 2021, pp. 1–5, doi: 10.1109/ICRITO51393.2021.9596330.

[8] S. Lyu and D. C.-T. Lo, “Fake news detection by decision tree,” in *Proc. 2020 SoutheastCon*, 2020, pp. 1–2, doi: 10.1109/SoutheastCon44009.2020.9249688.