Fake News Stance Detection Based On Deep Learning

Mrs.P. M. Logeshwari,
Assistant Professor,
Computer Science and Engineering,
Shree Venkateshwara Hi-Tech
Engineering College,
Gobi, Erode
loguvicky31@gmail.com

Mr.M.Guna,
Student,
Computer Science and Engineering,
Shree Venkateshwara Hi-Tech
Engineering College,
Gobi, Erode
gunamanoj2@gmail.com

Mr.R.Poovarasan,
Student,
Computer Science and Engineering,
Shree Venkateshwara Hi-Tech
Engineering College,
Gobi, Erode
pkpoovarasan1532@gmail.com

Mr.S.Sounthirarajan
Student
Computer Science and Engineering.
Shree Venkateshwara Hi-Tech
Engineering College
Gobi, Erode
sounthar2506@gmail.com

ABSTRACT—The proliferation of fake news on social media platforms has become a significant societal concern, influencing public opinion and undermining trust in information sources. Detecting the stance of news articles towards a given topic is crucial for assessing their credibility. This paper presents a novel approach to fake news stance detection utilizing a Multilayer Perceptron (MLP)-based deep learning architecture. We propose a neural network architecture that incorporates both text and metadata features to effectively capture the nuanced linguistic and contextual cues indicative of stance. Experimental results on a benchmark dataset demonstrate the efficacy of the proposed method in accurately classifying news articles into various stance categories, outperforming state-of-the-art baseline models. Our approach, leveraging MPLs, offers a promising avenue for combating the spread of misinformation and promoting media literacy in the digital age.

KEYWORDS- *Fake news, Stance detection, MLP, Multilayer Perceptron, Deep learning, Text analysis, Social media, Misinformation, Media literacy.*

INTRODUCTION

These days" fake news is creating different issues from sarcastic articles to a fabricated news and plan government propaganda in some outlets. Fake news and lack of trust in the media are growing problems with huge ramifications in our society. Obviously, a purposely misleading story is "fake news" but lately blathering social medias discourse is changing its definition. Some of them now use the term to dismiss the facts counter to their preferred viewpoints. The importance of disinformation within American political discourse was the subject of weighty attention, particularly following the American president election. The term 'fake news' became common parlance for the issue, particularly to describe factually incorrect and misleading articles published mostly for the purpose of making money through page views. In this paper, it is seeked to produce a model that can accurately predict the likelihood that a given article is fake news. Facebook has been at the epicenter of much critique following media attention. They have already implemented a feature to flag fake news on the site when a user see's it, they have also said publicly they are working on to distinguish these articles in an automated way. Certainly, it is not an easy task. A given algorithm must be politically unbiased – since fake news exists on both ends of the spectrum – and also give equal balance to legitimate news sources on either end of the spectrum. In addition, the question of legitimacy is a difficult one. However, in order to solve this problem, it isnecessary to have an understanding on what Fake News is. Later, it is needed to look into how the techniques in the fields of machine learning, natural language processing help us to detect fake news.

FAKE NEWS DETECTION

However, on the other hand, social media provides the ideal place for the creation and spread of fake news. Fake news can become extremely influential and has the ability to spread exceedingly fast. With the increase of people using social media, they are being exposed to new information and stories every day.
Misinformation can be difficult to correct and may have lasting implications. There for we need to make sure the news we read are correct and real so, there should be be fake news detection in our applications so that we don"t fall for those fake news spreading around us.

PREDICTIVE ANALYSIS

Predictive analytics uses historical data to predict future events. Typically, historical data is used to build a mathematical model that captures important trends. That predictive model is then used on current data to predict what will happen next, or to suggest actions to take for optimal outcomes. Predictive analytics has received a lot of attention in recent years due to advances in supporting technology, particularly in the areas of big data and machine learning. Predictive analytics is often discussed in the context of big data, Engineering data, for example, comes from sensors, instruments, and connected systems out in the world. Business system data at a company might include transaction data, sales results, customer complaints, and marketing information. Increasingly, businesses make data-driven decisions based on this valuable trove of information. To extract value from big data, businesses apply algorithms to large data sets using tools such as Hadoop and Spark. The data sources might consist of transactional databases, equipment log files, images, video, audio, sensor, or other types of data. Innovation often comes from combining data from several sources. With all this data, tools are necessary to extract insights and trends. Machine learning techniques are used to find patterns in data and to build models that predict future outcomes. A variety of machine learning algorithms are available, including linear and nonlinear regression, neural networks, support vector machines, decision trees, and other algorithms.

DEEP LEARNING

Deep learning is a machine learning technique that teaches computers to do what comes naturally to humans: learn by example. Deep learning is a key technology behind driverless cars, enabling them to recognize a stop sign, or to distinguish a pedestrian from a lamppost. It is the key to voice control in consumer devices like phones, tablets, TVs, and hands-free speakers. Deep learning is getting lots of attention lately and for good reason. It's achieving results that were not possible before. In deep learning, a computer model learns to perform classification tasks directly from images, text, or sound. Deep learning models can achieve state-of-the-art accuracy, sometimesexceedinghuman-levelperformance. Models are trained by using a large set of labeled data and neural network architectures that contain many layers. Deep learning achieves recognition accuracy at higher levels than ever before. This helps consumer electronics meet user expectations, and it is crucial for safety-critical applications like driverless cars. Recent advances in deep learning have improved to the point where deep learning outperforms humans in some tasks like classifying objects in images. While deep learning was first theorized in the 1980s, there are two main reasons it has only recently become useful: Deep learning requires large amounts of labeled data.

RELATED WORK

Although significant research has been conducted on detecting fake news, primarily focusing on textual data or uni-modal features, recent studies have introduced deep learning-based methods such as the ELD-FN approach. Unlike traditional approaches, ELD-FN not only addresses single-modal features but also incorporates sentiment analysis into textual information. State-of-the-art fake news classification methods generally fall into two categories: 1) thosefocusing on single-modality features, and 2) those considering multi-modality features.

EXISTING SYSTEM

Fake news detection is a difficult problem due to the nuances of language. Understanding the reasoning behind certain fake items implies inferring a lot of details about the various actors involved. We believe that the solution to this problem should be a hybrid one, combining machine learning, semantics and natural language processing. The purpose of this project is not to decide for the reader whether or not the document is fake, but rather to alert them that they need to use extra scrutiny for some documents. Fake news detection, unlike spam detection, has many nuances that aren't as easily detected by text analysis. Besides detecting fake news articles, identifying the fake news creators and subjects will actually be more important, which will help completely eradicate a large number of fake news from the origins in online social networks. Generally, for the news creators, besides the articles written by them, we are also able to retrieve his/her profile information from either the social network website or external knowledge libraries, e.g., Wikipedia or government-internal database, which will provide fundamental complementary information for his/her background check. Based on various types of heterogeneous information sources, including both textual contents/profile/descriptions and the authorship and article subject relationships among them, we aim at identifying fake news from the online social networks simultaneously. We formulate the fake news detection problem as a credibility inference problem, where the real ones will have a higher credibility while unauthentic ones will have a lower one instead.

PROPOSED SYSTEM

In this proposed method for detecting fake news harnesses the power of deep learning, specifically Multilayer Perceptron (MLP), to automatically discern deceptive or misleading information within textual content. In an era characterized by the rapid dissemination of information through digital platforms, the proliferation of fake news poses a significant threat to public discourse and societal well-being. Traditional approaches to combating misinformation often rely on manual fact-checking or rule-based algorithms, which are time-consuming, labor-intensive, and may struggle to keep pace with the sheer volume and complexity of online content. In contrast, our method capitalizes on the capabilities of MLPs, a type of artificial neural network renowned for its ability to learn complex patterns and relationships within data.

ABOUT THE MLP ALGORITHM

In fake news stance detection, the MLP (Multilayer

Perceptron) algorithm serves as a powerful tool for classifying the stance of news articles toward particular topics. It functions by analyzing labeled datasets, typically comprising news articles paired with their stances (such as "support," "deny," or "neutral") regarding specific subjects or claims. Before applying MLP, the data must be prepared. This involves acquiring a dataset with labeled examples, where each news article is associated with its stance toward a topic. Additionally, features need to be extracted from the text data. These features can range from simple word frequencies to more complex representations like TF-IDF scores or embeddings such as Word2Vec or GloVe. Once the data is prepared, the MLP model is trained. The neural network takes the extracted features of news articles as input and predicts their stances as output. During training, the model adjusts its parameters to minimize the difference between predicted stances and actual labels. Validation and tuning are crucial steps to ensure the model's effectiveness. Validation involves assessing the model's performance using a separate dataset to prevent overfitting or underfitting. Fine-tuning hyperparameters like learning rate, number of layers, neurons, and activation functions optimizes performance. After validation, the model is tested on a separate test dataset to evaluate its ability to accurately detect the stance of fake news articles. If the model performs satisfactorily, it can be deployed to make predictions on new, unseen news articles. MLP algorithms are effective in fake news stance detection because they can capture intricate relationships between input features (such as textual content) and output labels (stances). However, success depends on the quality and relevance of extracted features, as well as the size and diversity of the training dataset.

INPUT DATASET

The official website provided the benchmark dataset for Fake News Challenges. The FNC dataset has 2, 587 article bodies and 75, 385 tagged instances. These article bodies correspond to roughly 300 headlines, and there are five to twenty news items for every allegation. Among these headlines, Table illustrates that 7.4% are in agreement, 2.0% are in disagreement, 17.7% are in discussion, and 72.8% are unconnected. The claims pertaining to the bodies of the articles are carefully labeled. The labels' specifics are as follows:
•Agreed: The article body and headline are related.
•Disagree: The article body and headline have no connection.
•Talk about it: Taking it as impartial, the headline and article body somewhat match.
•Unrelated: The subject matter covered in the body andheadlineareunrelated. There are 49, 972, and 25, 413 instances in the dataset. for testing and training, correspondingly. The guidelines outlined for the FNC-1 challenge are the basis for this distribution of training and testing data. In the training set, there are 1, 648 headlines and 1, 683 article bodies. There are about 880 headlines and 904 article bodies in the test data.

PREPROCESSING

Pre-processing is a data mining approach that converts inconsistent and incomplete raw data into a format that is comprehensible by machines. The FNC- 1 dataset was used to carry out a number of text pre- processing operations. Arranged NLP techniques including stop word removal, stemming, tokenization, and converting text characters to lowercase letters were used, together with algorithms from Keras's library, to complete these tasks. Stop words are often used terms, such as "of," "the," "and," "an," etc., that are unimportant for this work and have very little meaning in the text. By eliminating the stop words, we shorten processing times and free up space that would have been occupied by the previously listed meaningless words. Words with comparable meanings may appear more than once in the text, such as games and games. If so, it works very well to reduce the words to a common fundamental form. This The NLTK's Porter stemmer method is implemented in an open-source manner for a procedure known stemming. Following the implementation of the aforementioned pre-processing procedures, there were only 372 terms remaining in the headlines. Each headline was divided into a vector of words using the tokenizer function from Keras library. Following the completion of the preprocessing, word/text is mapped to a list of vectors using word embedding (word2vec). Ultimately, 5,000 unigram terms found in headlines and article bodies are compiled into a dictionary.

TRAIN THE DOCUMENTS

Text summarization, an application of information retrieval, condenses extensive electronic collections on the internet into concise versions, retaining key information and meaning. It addresses the challenge of information overload by enabling users to efficiently access relevant data. This method, often applied to query-specific document summaries, utilizes similarity measures for effective summarization. Users can upload standard text files to this module, facilitating the collection and summarization of large news datasets. Such datasets, which list values for variables like text content, enable efficient data retrieval and analysis .

TEXT MINING

The initial phase involves gathering text documents stored in .TXT format. Following this, the document undergoes preprocessing, where redundancies, inconsistencies, and individual words are addressed, along with stemming. This prepares the documents for subsequent stages, which include:
1.Tokenization: The document is treated as a string, and each word is identified as a separate token, dividing the document into units.
2.Stop Word Removal: Common words such as "a," "an," "but," "and," "of," and "the" are eliminated from the text.
3.Stemming: This process aims to find the base form of words by identifying natural groups with similar meanings. Stemming methods, including inflectional
and derivational stemming, are employed, with   Porter's algorithm being a widely used option.

DOCUMENT TERM MATRIX CONSTRUCTION

This module enables the computation of term frequency (TF) and inverse document frequency (IDF). TF-IDF, an acronym for term frequency– inverse document frequency, is a numeric measure used in information retrieval to gauge the significance of a word within a document collection. It serves as a crucial weighting factor in information retrieval, text analysis, and user behavior modeling. The TF-IDF score of a word increases with its frequency in the document but is counterbalanced by its occurrence across the entire corpus, thereby mitigating the impact of commonly occurring words. Additionally, the module computes entropy and the probability of IDF. Entropy assigns greater weight to terms appearing less frequently across documents, while normalization corrects variations in document lengths and standardizes document vectors. The first step involves preprocessing the raw text data. This typically includes tasks like tokenization (splitting text into words or phrases), lowercasing, removing stopwords (common words like "and," "the," etc.), and possibly stemming or lemmatization to reduce words to their root forms. Once the text is preprocessed, it needs to be transformed into numerical vectors that deep learning models can process.

CLASSIFICATION

Classification is the core task of the training module, where the MLP algorithm learns to differentiate between genuine and fake news articles based on the extracted features. The classification stage involves the following steps: Model Selection: MLP, a type of feedforward artificial neural network, is chosen as the classification model due to its ability to learn complex patterns in data. The architecture of the MLP consists of an input layer, one or more hidden layers, and an output layer. Training: The MLP model is trained on a labeled dataset comprising genuine and fake news articles. During training, the model learns to adjust its parameters (weights and biases) through backpropagation, minimizing the error between predicted and actual labels. Hyperparameter Tuning: Hyperparameters such as learning rate, number of hidden layers, and activation functions are fine-tuned to optimize the performance of the MLP model. This process involves experimentation and cross-validation to identify the most effective parameter configurations.
Evaluation: The trained MLP model is evaluated on a separate validation or test set to assess its performance in terms of accuracy, precision, recall, F1 score, and other relevant metrics. The evaluation results provide insights into the model's effectiveness in detecting fake news.

BUILDING THE DATASET

Building a high-quality dataset is essential for training an MLP algorithm for fake news detection. This process involves the following steps: Data Collection: Genuine and fake news articles are collected from diverse sources, including online news platforms, social media, and fact-checking websites. Care is taken to ensure the representativeness and balance of the dataset. Labeling: Each news article in the dataset is labeled as genuine or fake based on reliable sources or fact-checking reports. Manual labeling may be supplemented by automated

methods such as keyword-based filtering or crowdsourcing. Data Augmentation: Data augmentation techniques such as paraphrasing, text synthesis, and data synthesis are employed to increase the diversity and size of the dataset. Augmentation helps prevent overfitting and improves the generalization ability of the MLP model. Dataset Splitting: The dataset is split into training, validation, and test sets to facilitate model training, hyperparameter tuning, and evaluation. The proportions of the splits are determined based on best practices and statistical considerations.

EVALUATION MODULE

The Evaluation Module is responsible for evaluating the performance of the trained MLP algorithm on unseen data to assess its effectiveness in detecting fake news. This module utilizes separate validation or test datasets that were not used during the training process to evaluate the model's generalization ability. Evaluation metrics such as accuracy, precision, recall, F1 score, and ROC-AUC (Receiver Operating Characteristic Area Under the Curve) are commonly used to quantify the performance of the fake news detection system. The Evaluation Module may also generate visualizations, such as confusion matrices or ROC curves, to provide insights into the model's strengths and weaknesses. Additionally, the module may conduct comparative evaluations with other fake news detection algorithms or baselines to benchmark the performance of the MLP-based approach.

TRAINING MODULE

The Training Module is responsible for training the Multilayer Perceptron (MLP) algorithm on the preprocessed textual data to develop a predictive model for fake news detection. This module utilizes a labeled dataset consisting of articles categorized as either fake or genuine to train the MLP algorithm. During training, the MLP learns to recognize patterns and relationships within the data that distinguish fake news from genuine news. The training process involves iteratively adjusting the parameters of the MLP, such as weights and biases, to minimize prediction errors and optimize model performance.

ARCHITECTUREDIAGRAM
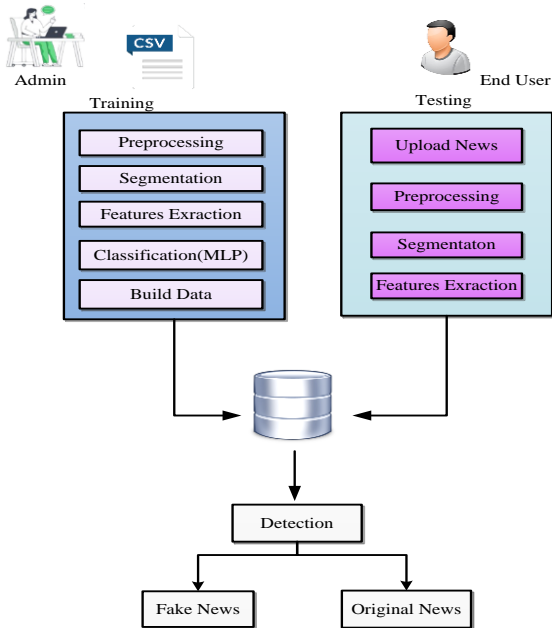
RESULT

In our final experimental phase, we trained an ensemble model combining CNN-LSTM on a dataset comprising 49,972 samples, followed by testing on 25,413 headlines and articles. The training process utilized a 2 GB Dell PowerEdge T 430 graphical processing unit, running on a machine equipped with 2x Intel Xeon 8 Cores clocked at 2.4GHz and 32 GB DDR4 Random Access Memory (RAM). Training involved pre-trained word embeddings and classification on the 'Fake News Challenge Dataset', taking approximately 3 hours for completion of epochs. In contrast, feature reduction techniques required 1.8 hours for computation. Comparisons were made among the outputs of the non-reduced feature set, PCA, and chi-square integrated into a CNN-LSTM architecture. Analysis suggests that PCA is more effective in significantly enhancing accuracy through severe dimensionality reduction. Our presented model outperforms others, achieving an accuracy rate of 97.8%. Furthermore, the average precision, recall, and F1-score for all classes are 97.4%, 98.2%, and 97.8% respectively, as detailed in Table 5, highlighting the statistical significance of our proposed model in distinguishing between fake and legitimate news.

BERT

BERT, short for Bidirectional Encoder Representations from Transformers, has been utilized in the FNC1 task, employing the fine-tuning method where all parameters are adjusted together, and a basic classification layer is appended to the pre- trained model. In this process, BERT predicts all masked positions independently, disregarding potential dependencies between them during training. This oversight leads to a reduction in the learning of certain dependencies simultaneously, resulting in inconsistency between pre-training and fine-tuning. Despite achieving 91.3% accuracy on the FNC-1 task,

BERT's F1-score falls significantly short when compared to both our model and the F1-scores of agree, disagree, and unrelated classes, as indicated in Table 7 for the CNN-LSTM model with k-fold cross- validation utilizing PCA.

XLNet

XLNet integrates bidirectional context while also avoiding independent predictions. It introduces a technique called "permutation language modeling," where tokens are predicted in a random order rather than sequentially. Built upon the Transformer XL architecture, XLNet surpasses BERT on 20 tasks, including document ranking, natural language inference, question answering, and sentiment analysis. It demonstrates improvement over BERT on the FNC- 1 task, achieving an accuracy of 92.1% and an F1- score of 76.0%

RoBERTa

An open-source language model known as Roberta (Robustly Optimized BERT Approach) was introduced in July 2019. In a study cited [67], the author developed a large-scale language model using transfer learning based on the Roberta-based deep transformer model. This model comprises 12 layers with 768 hidden units, each containing 12 attention heads, totaling 125 million parameters. To conduct transfer learning, they trained the model for fifty epochs and adhered to hyperparameter recommendations from [69], resulting in superior performance compared to both BERT and XLNet models. However, despite achieving an accuracy of 93.71%, which falls short of our model's accuracy, our proposed model, incorporating PCA and only one layer of CNN and LSTM, can achieve higher accuracy. We adjusted only a limited number of parameters, whereas Roberta entails tuning 125 million parameters precisely. Additionally, the computational costs escalate significantly with Roberta's 12 layers of 768 hidden units. Comparing F1-scores reveals that Roberta's performance on individual classes is inferior to our model's. This discrepancy may lead to inadequate performance even in agree and disagree classes, although the F1-scores for discussing and unrelated classes are almost identical.

CONCLUSION AND FUTURE WORK

This research introduces a stance detection model for identifying fake news, leveraging both headline and news body content, a departure from prior studies that focused solely on individual sentences or phrases. The proposed model integrates principal component analysis (PCA) and chi-square methods with CNN and LSTM architectures, where PCA and chi-square are employed to extract high-quality features fed into the CNN-LSTM model. Initially, the neural network processes the feature set without dimensionality reduction, followed by comparison with results obtained after applying dimensionality reduction techniques. PCA notably enhances the classifier's performance in detecting fake news by eliminating irrelevant, noisy, and redundant features from the feature vector. This approach yields promising outcomes,

achieving up to 97.8% accuracy, a significant improvement over previous studies. It's worth noting that dimensionality reduction methods can effectively decrease feature numbers while maintaining classifier performance. Future endeavors involve validating the proposed model's performance on larger datasets, exploring the potential superiority of tree-based learning over simplistic approaches, and analyzing various textual features and their fusion to enhance overall performance.

REFERENCES

1.T. Mihaylov, G. Georgiev, and P. Nakov, ''Finding opinion manipulation trolls in news community forums,'' in Proc. 19th Conf. Comput. Natural Lang. Learn., Beijing, China, Jul. 2015, pp. 310–314. [Online]. Available: https://www.aclweb.org/anthology/K15-1032

2.T. Mihaylov, I. Koychev, G. Georgiev, and P. Nakov, ''Exposing paid opinion manipulation trolls,'' in Proc. Int. Conf. Recent Adv. Natural Lang. Process., Hissar, Bulgaria, Sep. 2015, pp. 443–450. [Online]. Available: https://www.aclweb.org/anthology/R15-1058

3.T. Mihaylow and P. Nakov, "Hunting for troll comments in news community forums," in Proc. 54th Annu. Meeting Assoc. for Comput. Linguistics,

4.P. Bourgonje, J. Moreno Schneider, and G. Rehm, ''From clickbait to fake news detection: An approach based on detecting the stance of headlines to articles,'' in Proc. EMNLP Workshop: Natural Lang. Process. meets Journalism, 2017, pp. 84–89. [Online]. Available: https://www.aclweb.org/anthology/W17-4215

5.S. Vosoughi, D. Roy, and S. Aral, ''The spread of true and false news online,'' Science, vol. 359, no. 6380, pp. 1146–1151, Mar. 2018.

6.A. M. Michael Barthel and J. Holcomb. (2016). Many Americans Believe Fake News is Sowing Confusion. Accessed: Sep. 29, 2019. [Online]. Available: https://www.journalism.org/2016/12/15/many-americansbelieve-fake-news-is-sowing-confusion/

7.T Senthil Prakash, V CP, RB Dhumale, A Kiran., "Auto-metric graph neural network for paddy leaf disease classification" - Archives of Phytopathology and Plant Protection, 2023.

8.T Senthil Prakash, G Kannan, S Prabhakaran., "Deep convolutional spiking neural network fostered automatic detection and classification of breast cancer from mammography images" - Research on Biomedical Engineering,

9.TS Prakash, SP Patnayakuni, S Shibu., "Municipal Solid Waste Prediction using Tree Hierarchical Deep Convolutional Neural Network Optimized with Balancing Composite Motion Optimization Algorithm" - Journal of Experimental & Theoretical Artificial …, 2023

10.TS Prakash, AS Kumar, CRB Durai, S Ashok., "Enhanced Elman spike Neural network optimized with flamingo search optimization algorithm espoused lung cancer classification from CT images" - Biomedical Signal Processing and Control, 2023

11.C Aswath, T Prakash, P Kumari, N Thakur, R Sharma., " Effect of Gamma Radiation on Pollen Viability and Pollen Germination of Marigold Cultivar" - Think India Journal, 2019.

12.R. Senthilkumar, B. G. Geetha, (2020), Asymmetric Key Blum-Goldwasser Cryptography for Cloud Services Communication Security, Journal of Internet Technology, vol. 21, no. 4 , pp. 929-939

13.Senthilkumar, R., et al. "Pearson Hashing B-Tree With Self Adaptive Random Key Elgamal Cryptography For Secured Data Storage And Communication In Cloud." Webology 18.5 (2021): 4481-4497

14.Anusuya, D., R. Senthilkumar, and T. Senthil Prakash. "Evolutionary Feature Selection for big data processing using Map reduce and APSO." International Journal of Computational Research and Development (IJCRD) 1.2 (2017): 30-35.

15.Farhanath, K., Owais Farooqui, and K. Asique. "Comparative Analysis of Deep Learning Models for PCB Defects Detection and Classification." Journal of Positive School Psychology 6.5 (2022).