Comparing Machine Learning Methods in Fake News Detection

Krystal Dowling
914430
University of Melbourne
krystald@student.unimelb.edu.au

Charlotte Williams
University of Melbourne
914430
email

Coauthor
Affiliation
Address
email

Coauthor
Affiliation
Address
email

Abstract

The abstract paragraph should be indented ½ inch (3 picas) on both the left- and right-hand margins. Use 10 point type, with a vertical spacing (leading) of 11 points. The word **Abstract** must be centered, bold, and in point size 12. Two line spaces precede the abstract. The abstract must be limited to one paragraph.

1 Introduction

In a world of COVID-19 and an increasingly polarised and unstable political climate, fake news has become a real and credible threat to citizen liberty and health. It continues to disseminate insidious disinformation within a public sphere that increasingly turns to online social media for news [13][14][18].

The rise of fake news can be accredited to the fact that news can now be created and published online, thus bypassing the existing regulatory steps that bind traditional news media such as newspapers and television. Approximately 68

The failure of regulatory oversight stems from the overwhelming volume of published content produced daily by online sources. Conventional regulatory techniques to audit new content, such as human fact checkers, present an expense and time inefficiency that is unfeasible for companies contend with (reference?). A natural solution is through the use of machine learning techniques to automate the decision of truth and disinformation. Machines pose a scalable, efficient solution that may also present additional benefit through the reduction or elimination of human bias from news regulation.

This task of detecting fake news falls under the umbrella of natural language processing (NLP); more specifically, under text classification. NLP is a branch of machine learning which addresses the extraction of semantics and syntactic structure from human language (REFERENCE 2). It represents a complex problem; text is highly diverse and dimensional data (REFERENCE 3) and is not easily quantified.

In our report, previous studies and results will be taken into consideration to compare statistical machine learning with deep learning techniques. Methods used were those that have been proven to be successful in previous studies; thus for statistical models, a Naïve Bayes classifier was used as a baseline [6] and Support Vector Machine (SVM) as the best performing statistical machine learning model for fake news detection [6] [19]. Furthermore, two variants of deep learning were evaluated for this task: Convolutional Neural Networks (CNN) and Recurrent Neural Networks (RNN). CNNs are

cited as being appropriate for longer text classification [2], and they have been found to be effective in NLP problems as they are able to utilise the presence or absence of features, words in this situation, as a distinguishing factor [2].

Similarly for RNNs, they have been shown to be effective for sequence-based predictions such as NLP tasks in the past (REFERENCE 6). It is estimated that 20% of the meaning of a word comes from the context that the words occur in (REFERENCE 4). It is this notion that drove the selection of the context-orientated recurrent models for this task. A RNN's success in learning sequential data is a product of a chain-like structure underlying the model. However, the RNNs historically suffer from the vanishing gradient problem resulting in difficulties when learning long-term temporal dependencies in analysed sequences (REFERENCE 8). It is able to resolve the RNN's 'short-term' memory through the use of gating that regulates flow of information. The intent is to enable the LSTM to hold onto information that spans broader lengths of the input sequences. The model learns to retain that which is relevant to the prediction and to discard that which holds less predictive power.

In order to accurately judge each model and their relevance within the context of society, accuracy, precision, recall, and the generalisability of each model have been considered as evaluation metrics.

2 Data

Due to the lack of a single large and well-rounded dataset of labelled fake and real news, the research conducted as described in this report was performed on a combination of five datasets. Four of the five datasets were sourced from Kaggle, while the final dataset came from Signal Media. After examining the distribution of fake and real news from the Kaggle datasets, a clear imbalance towards fake news was found. To combat this, a sample of true news articles obtained from Signal media were also included, this meant the split between classes in the final dataset was almost even. As inferred from the word cloud shown in Figure X, the news articles revolve primarily around politics, specifically in the United States.

3 Method

3.1 Data Wrangling

From each of the previously listed datasets the article text and class label were extracted to be used in the classification models. Since some of the articles did not have a label, but instead the whole dataset was labelled as 'fake', a label column was created and assigned to these instances in the wrangling process. From this point the datasets were concatenated and treated as a single data frame.

3.2 Preprocessing

LUKE

3.3 Feature Engineering

3.3.1 TF-IDF

Term Frequency Inverse Document Frequency (TF-IDF) vector representations measure the importance of a word based on the number of times it appears in an article against the number of articles the word appears in.

3.3.2 Doc2Vec

LUKE

3.3.3 GloVe

CHARLOTTE

3.3.4 Neural Network Embeddings

Unlike one-hot encoded vectors that are often high-dimensional with uninformed mapping, word embeddings used in neural networks are continuous vectors from learned low-dimensional representations of discrete data that are able to capture relationships within language [12] [9] [15]. While these dense vector representations can be pre-trained, such as with GloVe explained above, they can also be learned within the neural network and thus tailored to training data [4]. The input is required to be integer-encoded for this, which was done using Tokenizer [4]. Then, once input into the embedding layer, the neural network learns optimal weights for these to minimise loss [4] [9].

3.4 Models

3.4.1 Statistical Machine Learning

The baseline model for this research was a Multinomial Naive Bayes. This model was produced using the TF-IDF embeddings, and the top 20,000 word features as determined by the chi-squared test.
Support Vector Machine (SVM) classification was explored as a statistical machine learner for fake news detection. The SVM aims to find a hyperplane which separates instances of different classes with the greatest margin and preferably no incorrect classification. This model was trained separately with three different embedding techniques; TF-IDF, GloVe and Doc2Vec. The TF-IDF vectors underwent feature selection using the chi squared test. After plotting the number of features against the model accuracy, the highest accuracy was achieved using the best 10,000 features. Since the attributes with the GloVe and Doc2Vec embeddings are the dimensions of a single vector, feature selection was not an option as removing any features would change the representation of the vectors. A grid search was then performed to determine the best kernel to fit the data; linear for TF-IDF, polynomial for GloVe and radial basis function for Doc2Vec, were found to produce the best results. The regularisation parameter was kept at its default of 1 to limit any misclassifications. The models were evaluated with the accuracy score and F1 score built into pythons sklearn metrics package.

3.4.2 Convolutional Neural Network

Multiple variations of CNNs were evaluated to get a full idea of their effectiveness in fake news detection. Convolutional neural networks have been found to be effective in NLP problems as they are able to utilise the presence or absence of features, words in this situation, as a distinguishing factor [2]. CNN's comprise of several convolutional layers (a convolution is a mathematical combination of two relationships to produce a third relationship [5]) applied over the input layer with nonlinear activation functions applied to results, in this case ReLU, the rectified linear activation function [1][12].

The standard model starts with an input layer and an embedding layer (explained above) [4]. Next, the convolutional layer applies different filters to the input, automatically learning the weights of its filters through back-propagation during training [3][7]. Next, a max-pooling layer is used to consolidate output while reducing the dimensional complexity and preserving salient information [5] [12] [16], it does this by only forwarding the maximum value from each feature map onto the next layer [7]. A flatten layer then converts 3-D data into a vector [5][16] and finally processed by a two fully-connected dense layers that connect the nodes of the previous layer to the next while applying an activation function [21] and an output layer [4].

This standard model was extended in the study to find the best performing CNN model, with iterations existing that utilised hyperparametrization to optimise values, and the creation multi-channel CNN with two and three channels of the standard CNN model.

3.4.3 Long Short-Term Memory

CHARLOTTE

4 Results

4.1 Support Vector Machines

The Multinomial Bayes baseline model performed at an accuracy of 83.10%, with an F1 score of 0.6431. As shown in Table 1, the SVM achieved its highest accuracy with the TF-IDF embedding. The SVM models with GloVe and Doc2Vec both performed worse than the baseline model, but very similarly to each other. The model with TF-IDF also had a significantly higher F1 score than the other models.

Table 1: SVM Results

Embedding	SVM accuracy	F1 Score
TF-IDF	0.8744	0.8495
GloVe	0.7712	0.7230
Doc2Vec	0.7796	0.7476

4.2 Convolutional Neural Network

The two main CNN models was a standard CNN set-up, and a multi-channel CNN model with three parallel channels comprised of the standard set-up; both sets of results are shown in Table 2.

Table 2: CNN Results

CNN Model	Accuracy	Precision	Recall	F1 Score
Standard	86.98%	89.44%	80.93%	84.97%
Multi-channel	89.75%	88.92%	87.14%	88.02%

5 Discussion

5.1 Convolutional Neural Network

Various convolutional neural network models were attempted in fake news detection, starting at a simple standard model and building on this model to the final best performing model; a multi-channel CNN.

The first CNN attempted was a standard CNN modified from Janakiev (2020). It had 128 filters, a kernel size of 5, and utilised ReLU in the convolutional layer, then had a max pooling layer and lastly a ReLU and Sigmoid function in the two final fully connected dense layers [7]. The ReLU, or rectified linear activation function, is a linear function that aims to introduce non-linearities to the network and overcome the vanishing gradient problem common in other activation functions [1][4]. ReLU outputs zero for non-negative inputs, and returns the direct input otherwise [1][4]:

$$g(z) = max(0, z).$$

The final two dense layers are fully-connected layers that connect the nodes of the previous layer to the next [21]; the first uses the ReLU function, and the second the Sigmoid function, which maps to a value in $[0, 1]$ and is thus particularly useful as a final layer for this model as it is predicting the probability of classes [17]. This standard CNN had its highest accuracy of 83.47% at a batch size of 16 and with 6 epochs. The precision was a respectable 83.59%, however the recall was lower at 75.66%, indicating the model is more inclined to make correct guesses of fake news rather than identify all fake articles.

However, this model had set a maximum length in padding of 100. By instead setting the maximum length for padding to the maximum article length, the CNN's accuracy increased significantly, to 86.98% (also with a batch-size of 16 and 6 epochs). The precision, recall, and F1-score were all higher than the previous model- 89.44%, 80.93% and 84.97% respectively.

This dramatic increase in the accuracy of the model was likely as previously the padding was cutting sequences larger than 100 [7]. From the descriptive statistics, this restricts the vast majority of the data; as 75% of articles contain over 255 words and the mean word length in articles was 657.83 words. Therefore, this indicates that increasing the text from the articles provides significant information to the CNN that allows it to vastly improve it's accuracy. This may be also partly due to the padding now functioning more effectively to highlight words on the edge of the input matrix and allow the padding to improve the robustness of the model [21].

Next, random search with cross validation was utilised as a hyper-parametrization method to optimise the parameters through running the model with various random parameter combinations [7]. The architecture of the model remains the same, with an embedding dimension of 100, and a maximum length and vocab size fit to the data. However, a list of different parameters were given for the number of filters (32, 64, 128), and kernel size (3, 5, 7) to optimise. There were 4 folds of cross validation fitted for 5 iterations.

The model peaked at 87.25% accuracy, with a batch-size of 16 and 6 epochs. This occurred when the number of filters was 64 and the kernel size was 3. This is a very slight increase from the standard CNN model, which had 128 filters and a kernel size of 5, likely indicating that the number of filters and kernel size do not have a huge influence on improving the accuracy of the model. However, the kernel size in the CNN for text classification defines the number of words, or n-grams, the convolution considers as a group [4]- thus the results indicating that variable kernel sizes can result in the best performing model could indicate that a variety of groupings of words are significant in building the model. This may be why the next model, a multi-channel CNN, performs best.

The next model created was a multi-channel CNN based on Yoon Kim's approach in his 2014 paper 'Convolutional Neural Networks for Sentence Classification' [8], which used multiple channels of the standard model with different sized kernels to be "processed at different resolutions or different n-grams (groups of words) at a time, whilst the model learns how to best integrate these interpretations", according to Brown (2017). The standard model differs slightly from the original created model, as it uses a dropout layer for regularisation to prevent initial batches of training data from disproportionately influencing the learning of the model [4].

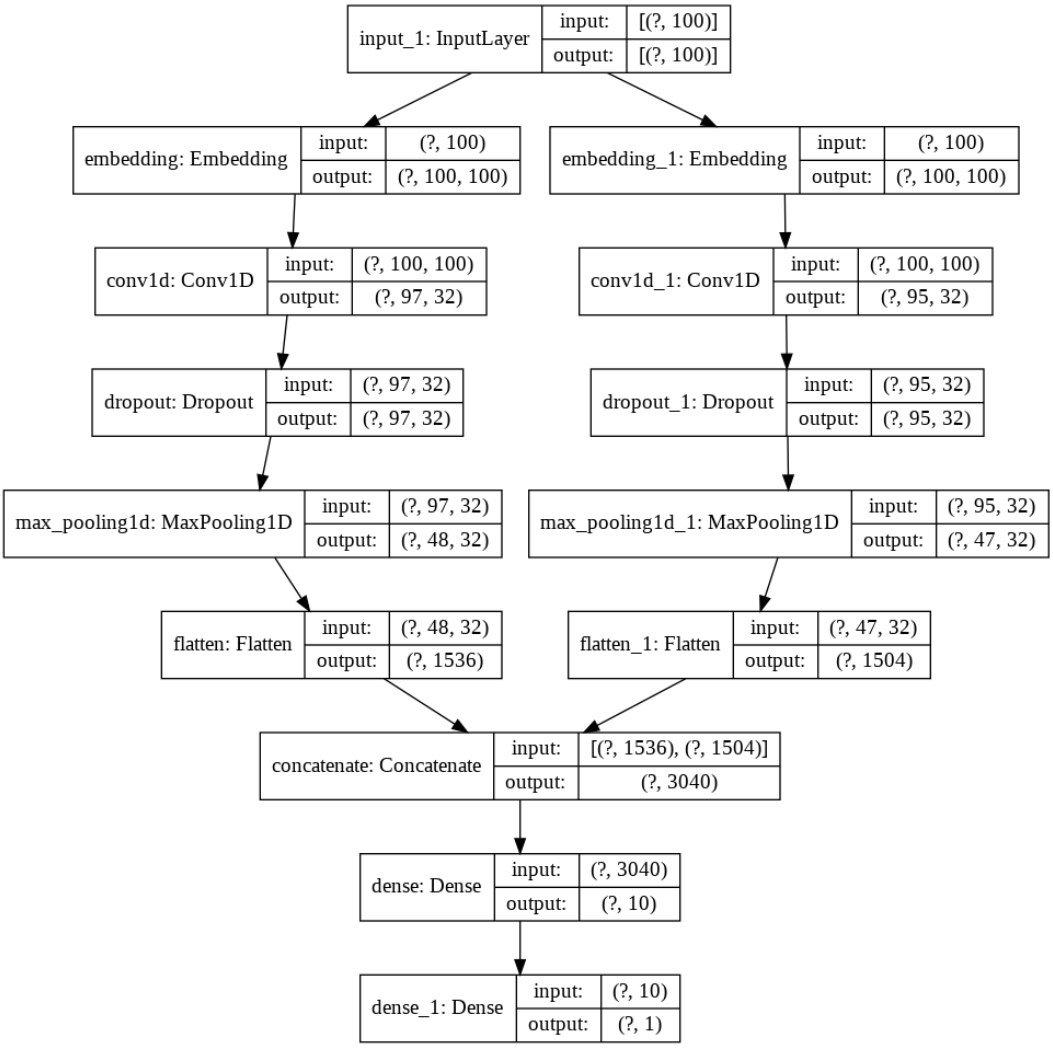
The 3-channel CNN was fit with 32 filters in each channel and kernel sizes of 4, 6, and 8 respectively. The model architecture is shown in figure 1 below. The highest accuracy was 89.75% for the multi-channel CNN. Like the standard model, the multi-channel CNN had a higher precision than recall; however, the difference was much smaller than in the standard model, with a precision of 88.92%, recall of 87.14%, and F1-Score of 88.02%. This indiates a fairly good model as they are all high values and the precision and recall are fairly balanced, as the F1-score indicates. A more precise model is also preferred over one with a higher recall for this task, as there is likely more associated risk with misidentifying a fake news story as true than incorrectly identifying a true news story as fake, as fake news tends to be more insidious and damaging when believed and encourages the spread of disinformation.

Despite this, the model is most accurate at low epoch values, finding its peak accuracy of 89.75% at 2 epochs and a batch size of 32. By analysing the models training and test accuracy and loss for large epoch values, such as at 25 epochs in figure 2, we can see that training accuracy reaches a value close to 100% fairly quickly (99.10% training accuracy for 2 epochs and 99.95% for 25 epochs). Further, the testing accuracy starts at a higher value and subsequently quickly decreases in early epochs before remaining fairly stable, if slightly decreasing, but with a seemingly large variance (?). Similarly, while the training loss decreases to almost zero, the testing loss continues to increase as the model fits more data.

This seems fairly indicative that the model is quickly over-fitting to the data, despite the added dropout layers to help prevent such [4]. This may be because the model is too complex for the problem and may be using too many filters on the data [21]. This further indicates that the model will likely no perform well when introduced to new data, as it is over-fitting to this dataset and is thus not very generalisable.

Despite this, the model still outperforms the SVM model explored above. While the standard model is fairly comparable to the SVM accuracy of 87.44%, the multi-channel model does give a higher accuracy, precision, recall, and F1-score. This could be due to the CNNs ability to utilise the presence or absence of features as a factor [2], and the SVM's inability to learn structure or order in text [20].

Figure 1: Architecture of Multi-Channel Convolutional Neural Network



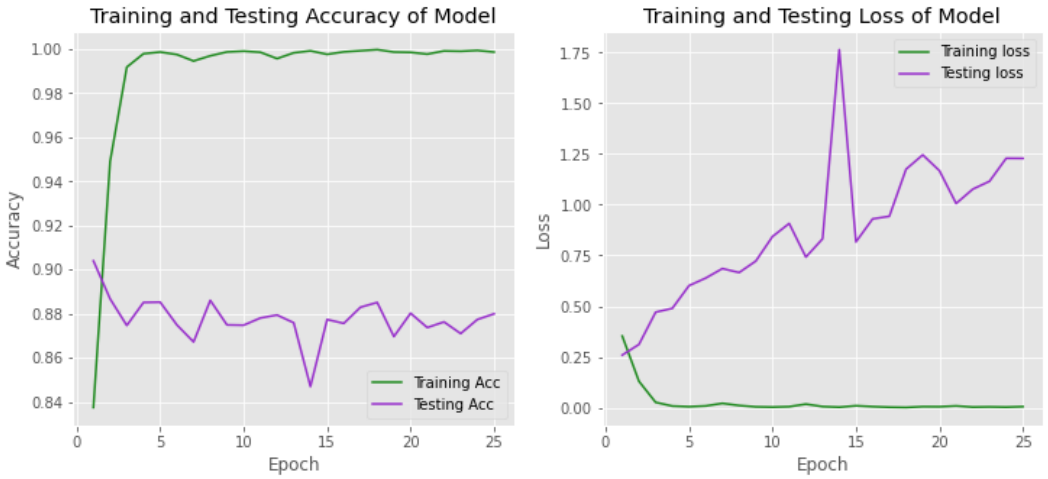
Further, the word embeddings in neural networks allow the CNN to recursively build more complex text representations that benefit the model [20], and large training datasets are also said to be difficult to implement for SVM models compared to CNNs [22].

6 Analysis between Models

Given that GloVe embeddings were so successful for the LSTM, they were also attempted on the CNN models. They resulted significant drop of test accuracy, 83.25%, for the multi-channel CNN, but gave a similar accuracy of 86.47%. This indicates that while the GloVe are comparable for the standard model and seem to function almost equally as well as the CNN embeddings, they are distinctly less accurate for the multi-channel CNN. This is likely as the embeddings in the multi-channel CNN without GloVe are better able to be tailored to the training data [4] and capture salient relationships within the text [9][15].

However, on the flip-side, this is likely part of what is contributing to the multi-channel CNN overfitting. By contrast, the GloVe embeddings for the multi-channel CNN do not result in overfitting; the training accuracy has a much slower rise to its peak and the test accuracy increases dramatically for the first 10 epochs before increasing only marginally for the next 40 epochs. Thus, this model may

Figure 2: Train and Test Accuracy and Loss of Multichannel-CNN



be more generalisable to new data, though without new data to test this hypothesis, a statement cannot be made either way. Similarly, the GloVe embeddings working so effectively on the LSTM-CNN model, make it more accurate, and is likely contributing to why it isn't overfitting like the CNN and is therefore more generalisable as a model.

The success of the multi-channel model is likely as the ability to perform parallel analysis on multiple n-gram features enhances the model's ability to interpret the different contributions of words to the semantic relationships of the text [11] [10]. However, the indication of these words having a significant semantic relationship relevant to their location in the text, is strongly suggestive that a recurrent neural network, which uses the sequential order of and location of words in fitting a model, might be highly effective on this data, which we verify in later analysis.

7 Future Research

- New data to test models ability to detect fake news - The current proposed model consists of a combination of existing models, so its limitations are clear, and to solve this problem, new techniques or designs of other architectures remain our priority in the future works. - that thing

8 Conclusion

References

[1] A. Amidi and S. Amidi. *Convolutional Neural Networks cheatsheet*. 2020.

[2] F.T. Asr and M. Taboada. "The global rise of "fake news" and the threat to democratic elections in the USA". In: *Public Administration and Policy: An Asia-Pacific Journal* 22.1 (2019), pp. 15–24. DOI: https://doi.org/10.1108/PAP-04-2019-0008.

[3] D. Britz. *Understanding Convolutional Neural Networks for NLP*. 2015. URL: http://www.wildml.com/2015/11/understanding-convolutional-neural-networks-for-nlp/. (accessed: 18.09.2020).

[4] J. Brownlee. *How to Develop a Multichannel CNN Model for Text Classification*. 2018. URL: https://machinelearningmastery.com/develop-n-gram-multichannel-convolutional-neural-network-sentiment-analysis/. (accessed: 13.09.2020).

[5] V. Choubey. *Text classification using CNN*. 2020. URL: https://medium.com/voice-tech-podcast/text-classification-using-cnn-9ade8155dfb9. (accessed: 20.10.2020).

[6] N. Conroy, V. Rubin, and Y. Chen. "Automatic Deception Detection: Methods for Finding Fake News". In: Oct. 2015. DOI: https://doi.org/10.1002/pra2.2015.145052010082.

[7] N. Janakiev. *Practical Text Classification With Python and Keras*. 2020. URL: https://realpython.com/python-keras-text-classification/#convolutional-neural-networks-cnn. (accessed: 06.10.2020).

[8] Yoon Kim. "Convolutional Neural Networks for Sentence Classification". In: *Proceedings of the 2014 Conference on Empirical Methods in Natural Language Processing (EMNLP)*. Doha, Qatar: Association for Computational Linguistics, Oct. 2014, pp. 1746–1751. DOI: 10.3115/v1/D14-1181. URL: https://www.aclweb.org/anthology/D14-1181.

[9] W. Koehersen. "Neural Network Embeddings Explained". In: *towards data science* (2018).

[10] P. Kumar. "An Introduction to N-grams: What Are They and Why Do We Need Them?" In: *XRDS* (2017).

[11] Z. Liu et al. *Multichannel CNN with Attention for Text Classification*. 2020. arXiv: 2006.16174 [cs.CL].

[12] M.M. Lopez and J. Kalita. *Deep Learning applied to NLP*. 2017. eprint: 1703.03091.

[13] G. Nyilasy. "Fake News in the Age of COVID-19". In: *Inside Business* (2020).

[14] A. Rajan. "Fake news: Too important to ignore". In: *BBC* (2017).

[15] R. Ruizendaal. "Deep Learning 4: Why You Need to Start Using Embedding Layers". In: *towards data science* (2017).

[16] S. Saha. "A Comprehensive Guide to Convolutional Neural Networks — the ELI5 way". In: *towards data science* (2018).

[17] S. Sharma. "Activation Functions in Neural Networks". In: *towards data science* (2017).

[18] K. Shu et al. "Fake News Detection on Social Media: A Data Mining Perspective". In: *SIGKDD Explor. Newsl.* 19.1 (2017), pp. 22–36. DOI: https://doi.org/10.1145/3137597.3137600.

[19] N. Smitha and R. Bharath. "Performance Comparison of Machine Learning Classifiers for Fake News Detection". In: *2020 Second International Conference on Inventive Research in Computing Applications (ICIRCA)*. 2020, pp. 696–700. DOI: 10.1109/ICIRCA48905.2020.9183072.

[20] R. Trusov. *Why would any one use Recursive Neural Nets for text classification as against SVM or Naive Bayes or any traditional statistical models?* 2016. URL: https://www.quora.com/Why-would-any-one-use-Recursive-Neural-Nets-for-text-classification-as-against-SVM-or-Naive-Bayes-or-any-traditional-statistical-models. (accessed: 22.10.2020).

[21] A.G. Walters. *Convolutional Neural Networks (CNN) to Classify Sentences*. 2019. URL: https://austingwalters.com/convolutional-neural-networks-cnn-to-classify-sentences/. (accessed: 25.10.2020).

[22] Z. Wang and Z. Qu. "Research on Web text classification algorithm based on improved CNN and SVM". In: *2017 IEEE 17th International Conference on Communication Technology (ICCT)*. 2017, pp. 1958–1961. DOI: 10.1109/ICCT.2017.8359971.