WARSAW UNIVERSITY OF TECHNOLOGY
Faculty of Mathematics
and Information Science

Introduction to Machine Learning

Project of
COVID-19 Fake News Detection

Submitted by:
Amir Ali
317554

01 January 2022

List of Contents:

1. Introduction

During the covid-19 epidemic, a large amount of data goes quickly throughout the world on the internet, particularly on social media platforms, where people from all over the world have constant and simple access to upload materials and posts. A significant amount of widely circulated news contains false information that has a harmful impact on readers' cognitive and psychological health.

Fake news has had violent consequences in the past and continues to do so now. Today, particularly during a pandemic, social media platforms are being utilized to spread misinformation at breakneck speed. To combat the fatigue of fake news, we should either shun news altogether or use tools such as machine learning.

2. Literature Review

"Disinformation Has no Religions"

"The transmission of COVID-19 reportedly linked to 5G mobile networks." "Position a half onion in the corner of your room to attract COVID-19 germs." "The sunlight keeps COVID-19 at bay"[1][2][3].

As per article [4], during the early phases of the pandemic, bogus news reports like these and others disseminated quickly on social media. The spread of false information was so widespread that officials coined the term "infodemic" to describe it.

Fake news isn't a new phenomenon. However, with the rise of social media, there has been a tremendous surge in interest in it in recent years. Concerns that misinformation spread by other countries may have impacted the Brexit referendum and the US presidential election heightened interest in 2016.

Fake news is thought to have a negative impact on people's behavior. Fake news, for example, has been claimed to influence people's inclination to wear a mask, get vaccinated, or follow other public health measures.

As stated in the article [5], several probable pathways suggest that misperceptions are likely to have increased during the COVID-19 pandemic, even though no previous studies have investigated how they evolved throughout the epidemic. Previous research, based on discrimination theory, shows that economic conditions are a primary source of discrimination

towards outside groups and minority populations. Most countries took considerable measures to reduce the rate of coronavirus infection.

Lockdowns, for example, and curfews have been introduced in various countries. Employees in "non-essential" occupations were laid off or furloughed, while those in "essential" occupations were closed. Several companies went bankrupt.

According to [6], during the COVID-19 pandemic, fake news has primarily focused on health-related topics (contagiousness, social distance, mortality rates, medications, masks, vaccinations, and so on), and partisanship has hampered efforts to contain the virus's spread.

Following article [7], we know that the rapid spread of bogus news during the epidemic has been dubbed a "second pandemic." In the context of the epidemic, some research has recently suggested a probable link between fake news and misperceptions. Exposure to social media has been linked to misunderstandings about basic facts concerning COVID-19, according to studies. These misunderstandings were linked to a poorer level of compliance with health-related measures.

3. Problem Statement

The circulation of fake news regarding COVID-19 is a very life-threatening fault that most people are not even aware of. Through this project, our desired goal is to distinguish and detect fake news from a given dataset of news articles. Through this, only the true COVID news will be spreading, through which the transmission of fake news will be affected tremendously.

One of the major focuses of this project is to correctly identify the type of news, whether it is fake or true. Although there are conditions that prevent our goal of distinguishing and detecting fake news from a given dataset of news articles. Some of the obvious conditions are noted below :

1. To increase the dataset with more recent news articles, we have to add articles that are already in broadcast, and correctly identify them as true or fake. Doing this is extremely gruelling as it is hard to distinguish them, as there is a likelihood that fake news is highly in circulation nowadays.

2. The error rate of the model is not zero, so still some fake news can be classified as true by the model.

To resolve the issue with adding current news articles to the dataset and to train the model accordingly, the one proposed method is to have reliable sources and have the people who have experienced the news first-hand have a voice for the particular news article. Even if there is a slight doubt whether the news is false, it must be classified as false as it can have a

massive outcome in the result of the training model. This solution is more ethical and will result in the situation moving closer to our goal.

In this project, we have taken into account the label value instead as partially false news is also considered false. Our project goal was to train a model to recognize the news article as true or false, with the maximum accuracy.

4. System Architecture

Figure 4.1 shows a Proposed Architecture for COVID-19 fake news detection of the entire process. The COVID-19 fake news data set is labeled as fake or true. This forms a dataset that is fed into a Term Frequency-Inverse document frequency (TF-IDF) vectorizer which transforms words into numerical features (NumPy arrays) for training and testing. The transformed dataset is split into training and testing subsets and fed into Multilayer Perceptrons (MLPs) and Support Vector Machine (SVM). All implementation was performed on Python 3.9 version, and we compare the result MLPs and SVM and use the model for prediction.

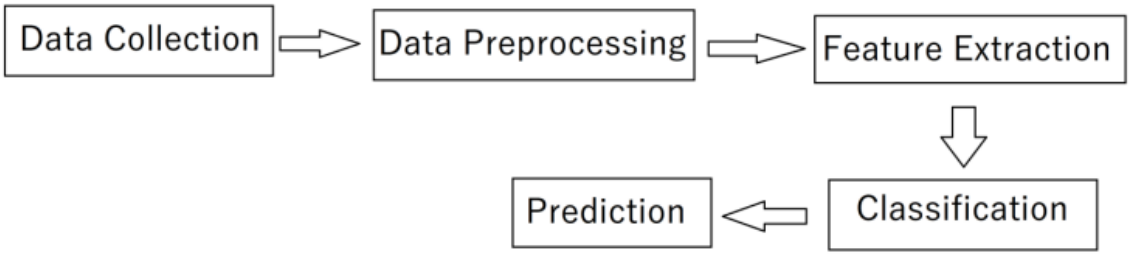


Figure 4.1: Architecture for COVID-19 Fake News Detection

4.1 Data Information

In this project, we have taken a dataset [8] with 3119 sets of news articles about the COVID-19 virus. The dataset consists of a collection of true and fake news related to COVID-19. The dataset consists of news between the period of December 2019- July 2020. The dataset does have an additional subcategory stating that the news is true, false, or partially false. But for the sake of clarification, the false and partially false news has a label value of 0, and the true news has a label value of 1.

4.2 Data Pre-processing

In any text mining problem, text cleaning is the first step where we remove those words from the document which may not contribute to the information we want to extract. In this part, we did data pre-processing. firstly, we did text cleaning with the help of Regular Expression and then we apply pre-processing operations like Tokenization to make a token of

each sentence, then remove stop words like "the", "a", "at" etc and then apply stemming to reducing a word to its word stem.

4.3 Feature Extraction

Once the dictionary is ready, we apply Term Frequency-Inverse Document Frequency (TF-IDF) model, and we take 5000 most frequent words from dictionaries for each COVID text of the whole dataset. Each word count vector contains the frequency of 5000 words in the whole dataset file.

4.4 Classifier Technique

After Feature Extraction we split our data for training and testing. And ratio for training and testing is 0.8 and 0.2 respectively. After that, we fed into classifier techniques Multilayer Perceptrons (MLPs) and Support Vector Machine (SVM), pipelines respectively. Below we explain the summary of both techniques that we use in our project.

4.4.1 Multilayer Perceptrons (MLPs)

An MLPs is a feed-forward network with one input layer, one output layer, and at least one hidden layer [9]. To classify data that is not linear in nature, it uses non-linear activation functions, mainly hyperbolic tangent or logistic function [10]. The network is fully connected, which means that every node in the current layer is connected to each node in the next layer. This architecture with the hidden layer forms the basis of deep learning architecture which has at least three hidden layers [11]. Multilayer Perceptrons are used for speech recognition and translation operations of NLP. Fig. 4.2 shows an MLPs with one hidden layer.[12]

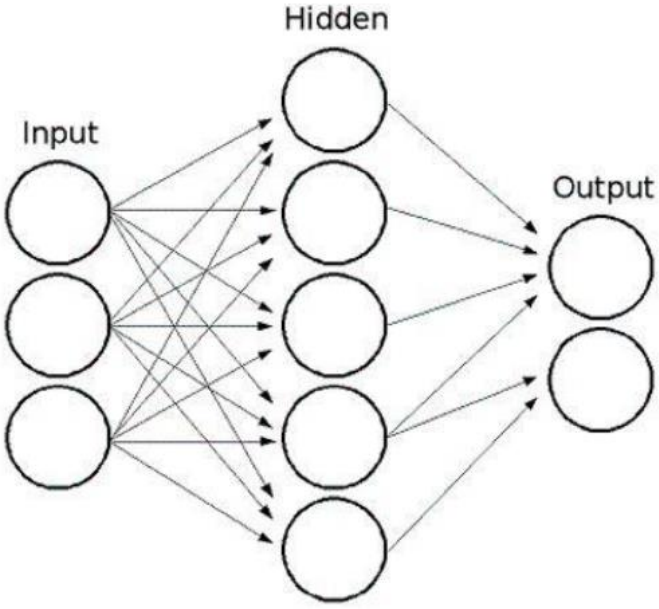


Figure 4.2: Multilayer Perceptrons with one hidden layer [13]

4.4.2 Support Vector Machine (MLPs)

A Support Vector Machine (SVM) is a discriminative classifier that can be used for both classification and regression problems. The goal of SVM is to identify an optimal separating hyperplane that maximizes the margin between different classes of the training data. In other words, given labeled training data (supervised learning), the algorithm outputs an optimal hyperplane that categorizes new examples to create the largest possible distance to reduce an upper bound. Supports Vectors are simply the coordinates of data points that are nearest to the optimal separating hyperplane that provides the most useful information for SVM classification. In addition, an appropriate kernel function is used to transform the data into a high dimension to use linear discriminate functions.[14]

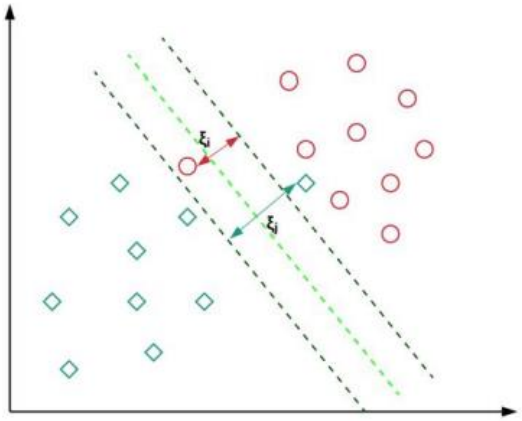


Figure 4.3: SVM with Kernel Trick [15]

5. Application Design

In this chapter, we develop a Model & Deploy It with Flask. Our model systems workflow is like this: Train offline -> Make model available as a service -> Predict online.

- A classifier is trained offline with Fake and True news.
- The trained model is deployed as a service to serve users.

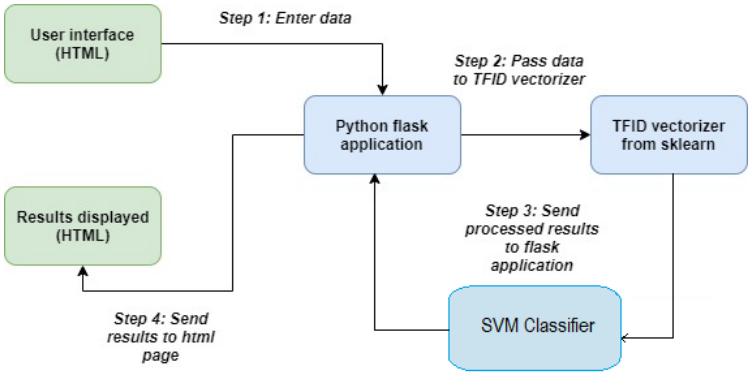


Figure 5.1: Application Workflow

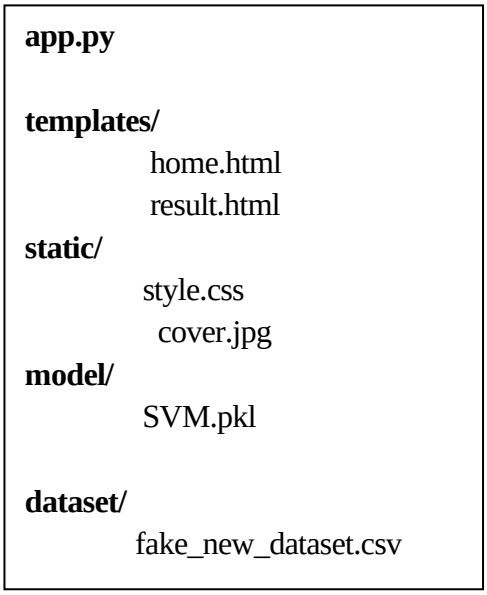
5.1. Building a Model

First, we will use the Covid-19 dataset to build a prediction model that will accurately classify the news. We test SVM and MLPs and SVM gave good results. Therefore we embed our best model (SVM) to build the application.

After training the model, it is desirable to have a way to persist the model for future use without having to retrain. To persist we make a pickle file of our model.

5.2. Turning Model into a Web Application

We develop a web application that consists of a simple web page with a form field that lets us enter a message. After submitting the message to the web application, it will render it on a new page which gives us a result of fake or true.



Application Folder File Directory

The sub-directory templates are the directory in which Flask will look for static HTML files for rendering in the web browser, in our case, we have two HTML files: home.html and result.html.

5.3. Running Procedure

Once we have done all of the above, we can start running the API by either double click `app.py`, or executing the command from the Terminal:

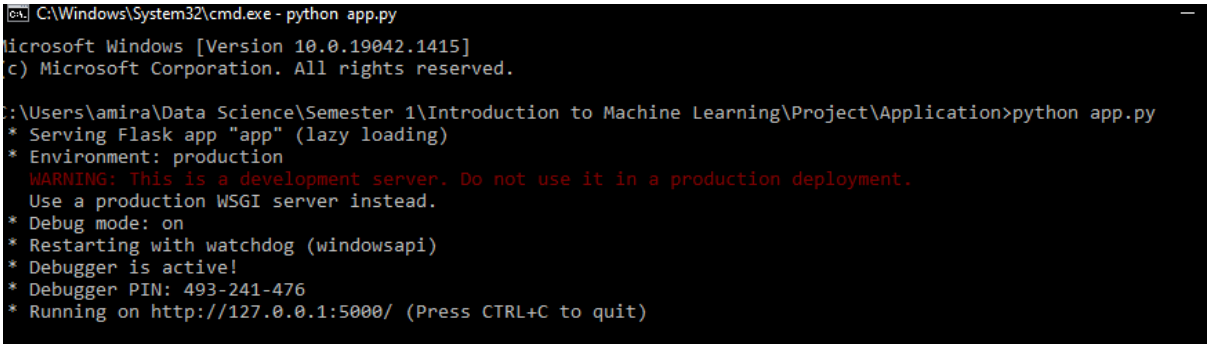


Figure 5.2: Command Execution

Now we could open a web browser and navigate to http://127.0.0.1:5000/, we should see a simple website with the content like so

Figure 5.3: COVID-19 Fake News Detection Website Page

Now we enter input in the comments form

Figure 5.4: Input In The Comments Form

After entering the input click the predict button now we can the result of our input.

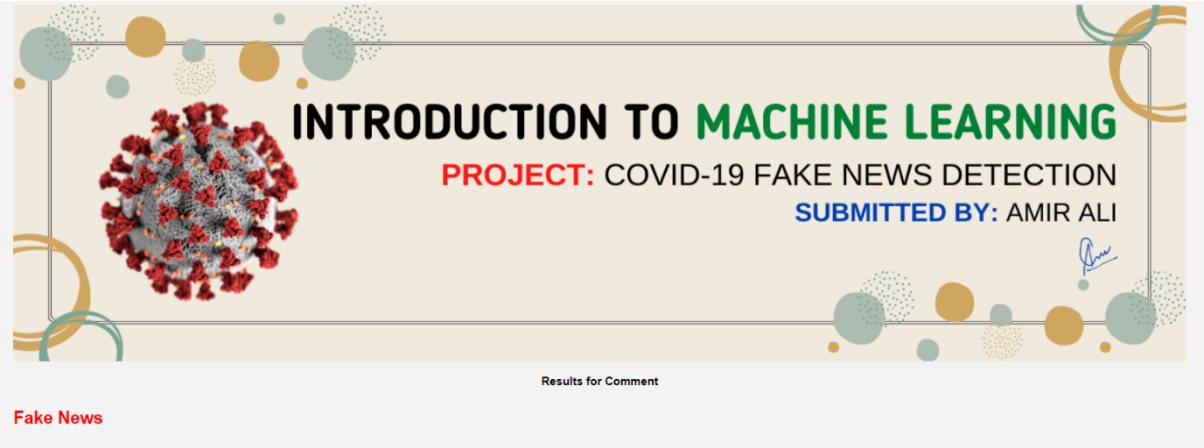


Figure 5.5: Fake News Example

6. Result Evaluation and Discussion

In this chapter, we evaluate our Result and also define the evaluation criteria to calculate the performances of our best classification model.

6.1 Evaluation Criteria

The confusion matrix was used to evaluate the classification models throughout the training process. The confusion matrix is a table that compares predicted and actual outcomes. It is frequently used to describe a classification model's performance on a set of test data.

Table 6.1: Confusion Matrix

Class	Predicted Negative	Predicted Positive
Actual Negative	TN	FP
Actual Positive	FN	TP

Important metrics were constructed from the confusion matrix in order to evaluate the classification models. In addition to the accurate classification rate or accuracy, other metrics for evaluation included True Positive Rate (TPR), True Negative Rate (TNR), False Positive Rate (FPR), False Negative Rate (FNR), Precision, F1 score, and Misclassification rate.

6.1.1 Accuracy or Correct Classification Rate (CCR)

AC is the proportion of the total number of predictions that were correct. AC is calculated as the number of all correct predictions divided by the total number of all predictions.

$$\text{Accuracy} = \frac{TP+TN}{TP+FP+TN+FN}$$

6.1.2 True Positive Rate (TPR)

The true positive rate (TPR) also known as sensitivity or recall is the proportion of positive cases that were correctly identified. TPR or Recall is calculated as the number of correct positive predictions divided by the total number of true positives and false negatives.

$$TPR = \frac{TP}{TP + FN}$$

6.1.3 True Negative Rate (TNR)

The true negative rate (TNR) also known as specificity is the proportion of negative cases that were correctly identified. TNR is calculated as the number of correct negative predictions divided by the total number of true negatives and false positives.

$$TNR = \frac{TN}{TN + FP}$$

6.1.4 False Positive Rate (FPR)

The false-positive rate (FPR) is the proportion of negatives cases that were incorrectly classified as positive. The false-positive rate is calculated as the number of incorrect positive predictions divided by the total number of false positives and true negatives.

$$FPR = \frac{FP}{FP+TN}$$

6.1.5 False Negative Rate (FNR)

The false-positive rate (FPR) is the proportion of negatives cases that were incorrectly classified as positive. The false-positive rate is calculated as the number of incorrect positive predictions divided by the total number of false positives and true negatives.

$$FNR = \frac{FN}{FN+TP}$$

6.1.6 Precision

PR is the proportion of the predicted positive cases that were correct. PR is calculated as the number of correct positive predictions divided by the total number of positive predictions.

$$\text{Precision} = \frac{TP}{TP+FP}$$

6.1.7 Error Rate (ER)

Also known as the 'Misclassification rate' is calculated as the number of false predictions divided by the total number of predictions.

$$\text{Error Rate} = \frac{FP+FN}{TP+FP+TN+FN}$$

6.1.8 F1 Score

The F1 score is calculated by taking the harmonic mean of precision and recall.

$$\text{F1 Score} = \frac{2 \text{ x Recall x Precision}}{\text{Recall+Precision}}$$

6.2 Result Discussion

In classification, a total of two algorithms implemented in Scikit-Learn were set as classifiers in detecting COVID-19 Fake news. The purpose of implementing two algorithms is to compare which model is best to use for this problem.

The classification of accuracy across two different classification algorithms such as Multilayer Perceptrons and Support Vector Machine using data proportion of 80:20 Ratio means 80% for training and 20% for testing.

And if we compare accuracy of both models then SVM gave much better result as compare to MLPs and accuracy result of SVM is 83% and MLPs is 78%.

Therefore, we will evaluate the results only for SVM. Below you see the confusion matrix visualization result of SVM.

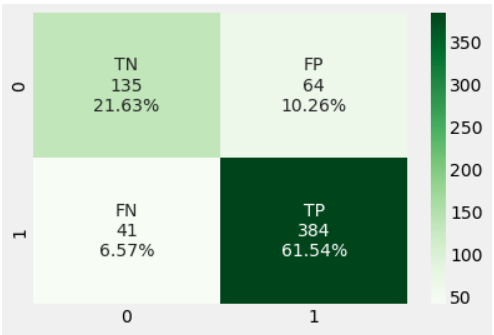


Figure 6.1: Confusion Matrix of SVM

Table 6.2 shows the final result that we evaluate on the basis of confusion matrix result

Table 6.2: Final Results

Classifiers	Accuracy	Precision	TPR	FPR	F1 Score	Error Rate	Specificity
SVM	0.8317	0.8571	0.9035	0.3221	0.8797	0.0.16	0.6783

Below you can see the visualization result of above table as well.

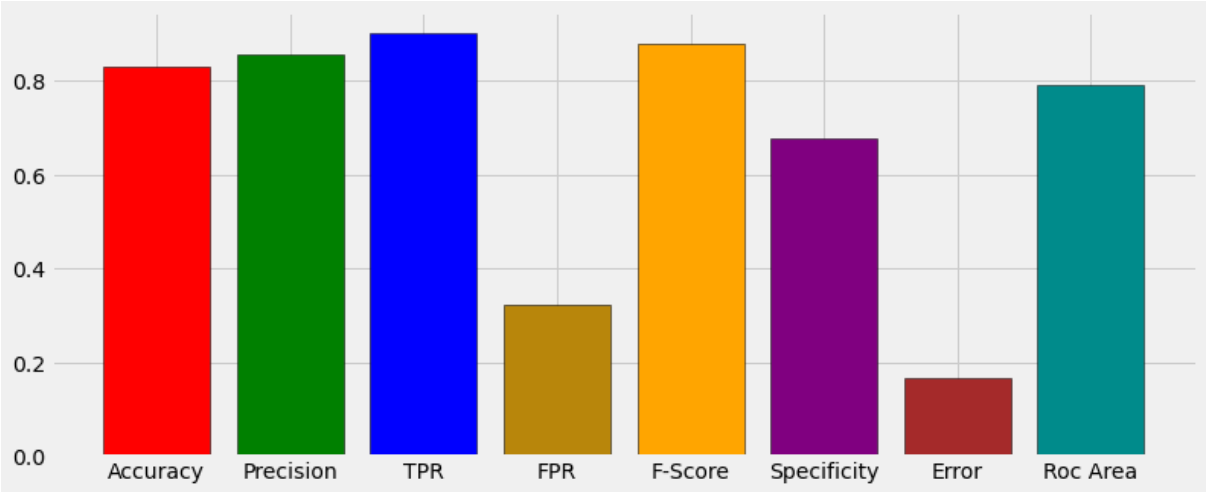


Figure 6.2: Visualization of Final Result

7. Conclusion

The goal of this project was to find capable methods and settings that could be used to help the detection of Cocid-19 Fake News. The error rate of the model is not zero, so still, some fake news can be classified as true by the model. In future we will enhance this work by implementing Temporal Convolutional Network (TCN) and Random Multimodel Deep Learning (RMDL) Techniques.

References

[1]https://factcheck.afp.com/http%253A%252F%252Fdoc.afp.com%252F9MD264-1

2]https://www.vox.com/recode/2020/4/24/21231085/coronavirus-5g-conspiracy-theory-covid-facebook-youtube

[3]https://eu.usatoday.com/story/news/factcheck/2020/03/30/fact-check-sunlight-does-not-kill-new-coronavirus/2931170001/

[4] https://www.hindawi.com/journals/cmmm/2021/5514220/

[5]http://www.sciencedirect.com/science/article/pii/S2211379721004034

[6] https://www.cesifo.org/en/publikationen/2021/working-paper/misperceptions-and-fake-news-during-covid-19-pandemic

[7] https://www.isglobal.org/en/healthisglobal/-/custom-blog-portlet/los-bulos-no-tienen-religion-las-fake-news-mas-comunes-sobre-las-vacunas-covid-19/3098670/0

[8] https://data.mendeley.com/datasets/zwfdmp5syg/1

[9] I. N. d. Silva, D. H. Spatti, R. A. Flauzino, L. H. B. Liboni, and S. F. d. R. Alves, Artificial Neural Networks A Practical Course, Springer International Publishing, 2017.

[10] O. Davydova, "7 Types of Artificial Neural Networks for Natural Language Processing," [Online]. Available: https://www.kdnuggets.com/2017/10/7-types-artificialneural-networks-natural language-processing.html.

[11] Y. Lecun, Y. Bengio and G. Hinton, "Deep learning," Nature, vol. 521, no. 7553, p. 436, 2015.

[12]https://www.academia.edu/41193006/An_Intuitive_Guide_of_Artificial_Neural_Network_with_Practical_Implementation_in_Keras_and_Tensorflow

[13] G. M and D. S. [Online]. Available: https://www.sciencedirect.com/science/article/pii/S1352231097004470.

[14]https://www.academia.edu/41133185/A_Deep_Level_Understanding_of_Support_Vector_Machine_with_Practical_Implementation_in_Scikit_Learn

[15] Support Vector Machines — Soft Margin Formulation and Kernel Trick available at https://towardsdatascience.com/support-vector-machines-soft-margin-formulation-andkernel-trick-4c9729dc8efe