

COVID tweets classification visualization and generation system

Erick Ruben Jaimes Curiel

Data Engineering

Universidad Politécnica de yucatán

Mérida, Yucatán, México

a11709069@upy.edu.mx

Ulises Lizandro Mis Pat

Data Engineering

Universidad Politécnica de yucatán

Mérida, Yucatán, México

a11709094@upy.edu.mx

Ariadna Elizabeth Moo Sosa

Data Engineering

Universidad Politécnica de yucatán

Mérida, Yucatán, México

a11709097@upy.edu.mx

Mariely del Rosario Nieves González

Data Engineering

Universidad Politécnica de yucatán

Mérida, Yucatán, México

a11709102@upy.edu.mx

Lisette Ruíz Peña

Data Engineering

Universidad Politécnica de yucatán

Mérida, Yucatán, México

st1809150@upy.edu.mx

Nilda Amira Yah Ucán

Data Engineering

Universidad Politécnica de yucatán

Mérida, Yucatán, México

a11709161@upy.edu.mx

Abstract—Covid-19 has made changes around the world, that is the reason why everyone is talking about it. Nowadays, thanks to technology which makes possible to know about the situation of Covid in each country of the world. The goal of this report is to present the process of collecting tweets related to Covid-19 in order to show how positive or negative people's comments are on Twitter. The project aims to build a model that uses deep learning technology to predict and classify different tweets, and generate new positive tweets in the case of negative ones. In order to build the model is carry up a Logistic regression. There are many different data mining techniques and methods for project management, but for the purpose of this report, the CRISP-DM method was chosen to structure the various stages of the project. Each stage contains activities and specifications to better understand the process and results.

Index Terms—COVID-19, Business Understanding, Data Understanding, Data Preparation, Modeling, Evaluation.

I. INTRODUCTION

The objective of this report is to present the process through the CRISP methodology in which information is stored about the steps that were taken to carry out a project seized by three different subjects, in which the only objective is the collection of tweets related to COVID-19 to show through a generative model trained how people have been reacting before the disease, within it you can find comments of positive or negative type also once classified tweets will develop a dashboard to visualize the data presented. At the same time another base model will be obtained in which through a dataset taken from Kaggle in which data about certain comments are found to know in the same way the same opinion of the people, which after applying certain preprocessing tasks will serve us to perform the training of the generative model, finally the observations and results of the models will be made along with the CRISP DM process which is a concise explanation of each procedure that was performed during the work delivered.

II. BUSINESS UNDERSTANDING

A. Background

Nowadays, social networks have become an important source of data generation and taking into account such a delicate but well-known issue worldwide as COVID-19, various comments (tweets) are generated daily on Twitter that refer to this term. Twitter is a famous social network and with which most people are familiar, that is why for the purposes of the project and in consideration of their requirements we made use of our official twitter accounts to use tweepy to collect tweets since we did not have the necessary data for the analysis and the requested processes.

Sentiment Analysis is a research area framed within the field of Natural Language Processing and whose main objective is the computational treatment of opinions, feelings and subjectivity in texts. For this project, a model will be trained in order to effectively predict the largest number of tweets, appropriately positive or negative.

B. Business Objectives

The principal objective for this project is to build a system classification comments and be able to visualize the obtained results.

- Train the model using the *toxic comments dataset*.
- Classify tweet text either positive or negative using the designed system.
- The system must return or create a positive comment given a negative one.
- Build a dashboard to visualize in real time the distribution of the comments and their prediction classification.

C. Business Success Criteria

In this case, an opinion is a positive or negative evaluation of a product, service, organization, individual, or any other type of entity that expresses a specific text. The system is

very useful for individuals and organizations to avoid or review negative comments when conducting public or webinar presentations on the subject. Knowing what people want to say, whether positive or negative, may be beneficial to certain organizations because they can use these comments to understand directors Or employees can improve to reduce negative comments (such as people's needs). From a business point of view, the possibility of classification is established as a success criterion The comment is correct. Another success criterion is to increase the number of positive comments on a given topic (in this case COVID19), accept it in the best way and try to generate Positive effect.

D. Inventory of Resources

We know the importance of data quality for models and predictions, based on the fact that we use the following sources as the main source of collection and consultation: 1) Twitter, because the services it provides are basically public. Freedom of expression allows users to express their thoughts and think freely, and, from the twitter user's profile, A lot of interest can be determined about them. 2. Scikit-Learn, which provides simple and effective tools for predictive data analysis, accessible to everyone, and can be reused in various situations. Basically its documentation is helpful because we use some machine learning libraries, and this source helps to use them correctly. 3. Kaggle, where was taken the "Toxic Comment Data Set".

As software resources, it was defined the use of python as main programming language, python includes libraries for data preprocessing and machine learning which are helpful for our building of the model. Data for database was collected using the twitter API (tweepy) which was stored in a MySQL database.

- Twitter account to get the key access and tokens
- Install mysql server
- Collect tweets data On the other hand, as hardware resources we used a laptop with the following characteristics:
 - Model: Inspiron 15 3000 series
 - Processor: Intel(R) Core(TM) i7-7500U CPU @ 2.70GHz
 - Storage: 120 gb ssd
 - RAM memory: 15gb ram ddr4 2133 mhz.
 - Operating System: Ubuntu

E. Requirements(Assumptions and Constraints)

Based on the requirements for the model, prediction and classification of the tweets as well as for their visualization, some of the restrictions that were had were the values of the location of the user who generated the tweet, which prevented us from knowing the location from the tweets. tweets, this resulted as an effect on the requirements of the visualizations.

F. Technologies Benchmarking

1) *Programming Languages:* A programming language is made up of a series of symbols that serves as a bridge

that allow humans to translate our thoughts into instructions computers can understand.

- Python

Python is one of the most popular languages used by data scientist. Python has become widely used for several reasons. It prioritizes readability, it's dynamically typed and it sports intuitive syntax, which makes it relatively easy to learn and use. It can be used to predict outcomes, automate tasks, streamline processes, and offer business intelligence insights. Python includes high-level data structures, dynamic typing, dynamic binding, and other features, making it suitable for complex application development [1]. Python is considered to be ideal for general purpose tasks like data mining and big data facilitation.

2) *Frameworks:* Both frameworks and libraries are code written by someone else that is used to help solve common problems. For this project we considered to make use of different frameworks.

- Keras

Keras is a high-level DL library written using Python; it runs on top of an ML platform known as TensorFlow. It is effective for rapid prototyping of DL models. Keras offers utilities for compiling models, graph visualisation and dataset analysis. Further, it offers prelabelled datasets that can be imported and loaded directly. It is user-friendly, versatile and suited for creative research [2].

- TensorFlow

TensorFlow is a popular Python framework for machine learning and deep learning, which was developed by Google. It's the best tool for tasks like object identification, speech recognition, and many others [3]. It helps in working with artificial neural networks that need to handle multiple data sets.

- Tableau

Tableau is considered as one of the best Business Intelligence and data visualization tools and has managed to top the charts quite a few times since its launch. The most important quality of this tool is that it makes organizing, managing, visualizing and understanding data extremely easy for its users.

3) Libraries:

- Scikit-learn

The Scikit-Learn framework includes ML algorithms like regression, classification, and clustering, among others. You can use it with other frameworks such as NumPy and SciPy.

- Numpy

NumPy is one of the most used libraries for tasks involving modern scientific computations and evolving yet powerful domains like Data Science and Machine Learning. The two vital benefits that NumPy has to offer is the support for powerful N-dimensional array objects and built-in tools for performing intensive mathematical as well as scientific calculations. Other impressive features

of NumPy include the use of an optimized C core for delivering high performance, interoperability with various computing platforms and hardware, and ease of use.

- Pandas

Pandas is another popular high-performance Python library that is being widely used today for solving modern Data Science and Machine Learning problems. By offering developers access to flexible yet extremely responsive data structures for working with time series and structured data along with the stack of other vital features, Pandas aims to become the best data analysis tool available for solving real-world problems.

- Matplotlib

It is a Python 2D plotting library which produces publication quality figures in a variety of hard copy formats and interactive environments across platforms. Matplotlib can be used in Python scripts, the Python and toolkits.

G. Risk and Contingencies

1) Risk and Control:

- Schedule

Each one of the activities proposed in the project plan must be followed in a timely manner to fulfill each one of them. The risk is reflected in the delay of activities in terms of time.

- Equipment

The aforementioned requirements of the computer equipment are essential for a good execution of the model, otherwise there is a risk of errors during execution.

2) Risk Avoidance:

- Resources

Sources must be reliable. Weak data can cause risks in model training.

- Procedures

Risks regarding processes, when one process depends on another, the proper functioning of this process is essential. Training and capabilities: Taking into account that the text of the tweet may contain grammatical errors, proper cleaning is important, otherwise it may result in a bad prediction.

H. Terminology

I. Costs and Benefits

For the purposes of the project, the data we need to work with is: 1. The toxic comment database that is provided free by Kaggle for download. 2. The database of tweets that will be collected using tweepy, free access API using the respective keys and tokens. In terms of cost, the only thing invested was the time to carry out the due processes, since free sources were used, the benefit they offer is greater since we obtained what was necessary without complications.

J. Data Mining Goals

From the perspective of data mining the objective is to create a system which be able to predict and classify comments either positive or negative to feed the model in order to

produce a new positive comment given a negative comment by making use of the encode decode generation technique are to create a system which classifies toxic comments and creates a non-toxic comment using the encode decode generation technique. The data mining goals are listed below:

- Distinguish and Classify the comments either positive or negative.
- System be able to re-produce a positive comment given a negative one.
- Visualize the obtained results in a dashboard.

K. Data Mining Success Criteria

From the point of view of data mining the success criterion are the possibility of classifying the comments with 90 accuracy is established as a success criterion in such a way that the system distinguish and classifies comments either positive or negative. Another success criterion is that the system can produce a new positive comment given the negative comment with a good level of understanding, logic about what it want to transmit to the public.

L. Project Plan

The project is defined as being completed within 14 days, from November 19th to December 2nd. During this time, the 5 stages of the project will be executed in different time periods, given the difficulty and the time required for training Models, assessments and obstacles. Business understanding (1 day): At this stage, we focus on understanding the project goals and requirements, computing resources and requirements.

- Data understanding (1 day): At this stage, we define the method of data collection, the data used in each step of the project, and verify the quality of the data.
- Data preparation (2 days): This part is for preprocessing Among all the data, data used for classification and text generation.
- Modeling (5 days): In this section, we make assumptions about the models that can provide good performance on the data.
- Evaluation (3 days): Once we have selected the models and trained the data, we will verify the performance of these models.
- Deployment (2 days): In the deployment, we generated the final document for the project as a visualization based on the data obtained for the classification and text generation model.

III. DATA UNDERSTANDING

A. Initial Data Collection Report

For this project we used two types of data but with the same context, in first place a dataset for fitting the model classification of the tweets sentiments, called Toxic Comments by Jigsaw and in the other hand it was taken a dataset from a social network called Twitter, in which we had to make certain requirements as a first instance we had to request a developer account, (which was obtained by two people from our team), from there to collect the data we used the Twitter

API where we had to request the access tokens such as the secret key, bearer token ,secret token or API key to call them from Jupyter Notebook, after obtaining all the tokens, the third step was to create a class listener that basically downloads the Twitter message exclusively only for the Tweets that contain the word.

The third step was to create a class listener that basically downloads the Twitter message exclusively for the Tweets that contain the COVID-19 tracking word, in that way 1000 tweets were obtained in real time, and then saved and preprocessed.

B. Data Description Report

The first data called Toxic comments has 159,571 comments from registrants, which were classified into six tags as toxic, severely toxic, insulting, threatening, obscene and identity hatred, where in each tag contains values of 1 and 0, making 0 in all tags a positive sentiment of the Tweet and 1 making the sentiment negative in the Tweet, in other words there are 144277 positive sentiments vs 15294 negative sentiments, furthermore the dataset has eight columns which are:

- id: Contains an unique id for each comment.
- Comment text: Store the comments done by wikipedia users.
- Toxic: Contains 0 if the comment is not toxic or 1 if is a toxic comment.
- Severtoxic: Determine the comment as 0 if the comment is not severe toxic or 1 if is a severe toxic comment.
- Obscene: Determine the comment as 0 if the comment is not obscene or 1 if is a obscene comment.
- Threat: Determine the comment as 0 if is non-threatening comment or 1 if is a threatening comment.
- Insult: Determine the comment 0 if does not contain bad words. 1 if it is a comment which contain some insults.
- Identity hate: Determine the level of the hate as 0 if is not considered a hateful comment or 1 if it is a real hateful comment.

	id	comment_text	toxic	severe_toxic	obscene	threat	insult	identity_hate
0	0000997932d777bf	ExplanationWhy the edits made under my usern...	0	0	0	0	0	0
1	0001030d9c6b60f	D'aww! He matches this background colour I'm s...	0	0	0	0	0	0
2	000113f07ec002fd	Hey man, I'm really not trying to edit war. It...	0	0	0	0	0	0
3	0001b41b1c6b637e	"nMoren! can't make any real suggestions on ...	0	0	0	0	0	0
4	0001d958c54c6e35	You, sir, are my hero. Any chance you remember...	0	0	0	0	0	0

Fig. 1. Dataset from Kaggle repository

Also, it could be considered that the Text dataset, contains 30588 rows and eight columns where only seven columns of numeric type, in which six of them would be considered columns with boolean values and the last one, the eighth column is considered of object type since it is only the text made by the users.

In the other hand, with the dataset of the Twitter, it could said that the data originally proposed was taken from the Twitter API explained in the collection section, for this reason the dataframe contains a seven columns which are:

- Tweet Id: Identification of each ID.
- Tweet Date: Date that the Tweet was done.

- Follower Count: the number of the account's followers.
- Account Verified: If the account is verified or not.
- Favorite Count: The number of the favourites counts.
- Retweets: The number of retweets of the tweet.
- Tweet Text: The column has the content of the tweet done by Twitter users.

	Tweet Id	Tweet Date	Follower Count	Account Verified	Favorite Count	Retweets	Tweet Text
0	146647224237865556	2021-12-02 18:20:02	10696	False	0	0	LIVE at 12:30 PM https://t.co/YuPERDGXhN@D...
1	1466472100139577358	2021-12-02 18:19:29	2387	False	0	0	@heatherhiv not sure he realizes the only "Rea...
2	1466472082787733506	2021-12-02 18:19:24	267	False	0	0	Unless Biden's plan calls for expediting @US F...
3	1466472071836413957	2021-12-02 18:19:22	6	False	0	0	@Fia_Pol @GovRonDeSantis @ByJasonDelgado Too b...
4	1466471973383462919	2021-12-02 18:18:58	3338	False	0	0	#No10ChristmasParties looks to be really upset...
...
994	1466434378188439562	2021-12-02 15:49:35	135482	True	0	0	The legislative Fiscal Committee signs off on ...
995	1466434350044561415	2021-12-02 15:49:28	1032	False	0	1	Two years in, and we have how many 'vaccines'?...
...
...	@SpikedPuppet See that's disgraceful.

Fig. 2. Dataset from API Twitter

To conclude this section, the dataset has 1000 rows and seven columns in which the tweets are stored, in which four of them are considered as numerical variables meanwhile two five column is a boolean variable and the other two columns are considered as date variable and object type.

C. Data Quality Report

After performing an initial scan, it can be confirmed that all the data is complete, because in the Toxic comments dataset there are no null, NA or outliers values that will be affected in the model training classification and in the Tweets dataset collected from API Tweet, since everything is text-based. Completed, there are no null values or out of range, but the main problem since they are comments made by people, the quality of the text is extremely low because it contains a lot of noise in between and also does not contain the column defining the type of sentiment per Tweet, so as not to affect the model generator to be implemented later, it has been decided to work with a kind of text cleaning and also through a sentiment analysis tool to give the type of reaction that has the Tweet through the polarity of the same.

IV. DATA PREPARATION

A. Dataset Deescription

The toxic database has 159,571 user comments collected from Wikipedia talk pages. These comments were annotated by human raters with the six labels 'toxic', 'severe toxic', 'insult', 'threat', 'obscene' and 'identity hate'. Comments can be associated with multiple classes at once. The comments that has zero in all the sentiments, they are positive comments.

- id: This field is the type object which contain an unique id for each comment.
- Comment text: This field is the type object which comments done by wikipebias user are stored.
- Toxic: This field is the type integer which contains 0 if the comment is not toxic or 1 if is a toxic comment.

- **Severtoxic:** This field is the type integer which contains 0 if the comment is not severe toxic or 1 if is a severetoxic comment.
- **obscene:** This field is the type integer which contains 0 if the comment is not obscene or 1 if is a obscene comment.
- **threat:** This field is the type integer which contains 0 if is non-threatening comment or 1 if is a threatening comment.
- **insult:** This field is the type integer which contains 0 if does not contain vulgar words. 1 if it is a comment which contain insults.
- **identity hate:** This field is the type integer which contains 0 if is not a hateful comment or 1 if it is a hateful comment.

For the tweets that were extracted from twitter have the following metadata suah as id, date-time, text, user, longitud and latitude.

1) *Data selection:* To perform the data cleaning, is necessary to define the main features in both datasets, for this in the first Toxic comments dataset as only wanted to work with the text and the sentiment had, so all the columns was taken with the sentiment set, and determined only one type of sentiment Toxic or Not Toxic through a function where if all the labels would have 0, then the text was equal to 0-Not toxic, while if in a label the text contained 1, then the Text was taken to be 1-Toxic, 3.

	comment_text	Toxic
0	Hey, I failed to notice this message previousl...	0
1	":::Please dont keep leaving your useless tral...	0
2	"\n\nSlash ""/"" and backslash ""\\" are the h...	0
3	"\nMaybe there's a better way to do it... Not ...	0
4	"\n\nThank you for your view. However, could ...	0

Fig. 3. Feature selection (Text column)

In the other dataset, only the last column of the dataset called Text was chosen, 4.


	Text
0	LIVE at 12:30 PM  https://t.co/YuPERDGrXH @D...
1	@heathergtv not sure he realizes the only "Rea...
2	Unless Biden's plan calls for expediting @US_F...
3	@Fla_Pol @GovRonDeSantis @byJasonDelgado Too b...
4	#No10ChristmasParties looks to be really upset...

Fig. 4. Feature selection (Text column)

The reason of chose this kind of fields in both dataset is because the feature will be used in sentiment analysis with the classification model and specifically for the generative model.

2) *Data cleaning:* In order to have good results in the two predetermined models, the first thing to be done was to did a task of preprocessing, in which the main objective sought is to clean and normalize the information, since within the texts extracted from the network and the Kaggle repository there are spelling errors, repetition of characters, mixture of upper and lower case, links, between others. To perform the objective first was to applied the cleaning of the data in the Tweet dataset, where a function called TweetCleaner was used, in which there were certain sub-functions exclusively for:

- Removing special characters such as numbers, hastags, @ or links, and make the Tweets have the same writing, in this case all are in lowercase this was done with the cleanString function.
- Elimination of empty words known as prepositions, pronouns, conjunctions, for the simple fact that do not provide any extra information, to achieve this we applied the function of stopwords and tokenizer taken by nltk library.
- The Stemming function was applied in order to have the text with the morphological normalization in a more accurate way compared to the lemmatization function. As have a final version, all the outputs of the main function were saved in a dataframe having the following columns as:
 - Text: The Tweets in their original format.
 - Cleaned Text: Tweets with cleaning but without normalization.
 - Steammed Text: Tweets with cleaning and normalization.
 - Polarity: The sentiment 0 or 1 that is attributed to the Tweet, in this case 0 is non-toxic and 1 is a toxic sentiment.

In spite of having the data ready to be able to attach it to the model, we would be missing the main factor called as the target of the model, which was defined later thanks to from *nltk.sentiment*. *vader import SentimentIntensityAnalyzer* (a tool taken from the nltk library which is based on a computational process that determines through certain words what type of sentiment the text expresses whether neutral, positive or negative), it was attached to the dataframe through a for-loop, so our dataframe at the end contained four columns and 1000 rows. Even having the last dataset with the data cleaning and normalization, it was necessary a feature selection, in which only the steammed text and polarity columns were selected, because the steammed text has all the Tweets done by the users and the polarity column has the sentiment of the the people related to the Twitter.SO, for the models, it can be can simplify that the latest version of the data is a dataset with only two columns and 1000 rows, 5.

In the other hand, with the Text comment dataset only text cleaning and text normalization was applied easier because in this dataset it was not a lot of noise. In order to have a better text again the Tweetcleaner function was applied but in this ocasion is focused on:

	Steammed_Text	polarity
0	live pm media brief wisconsin covid respons co...	0
1	sure realiz real america done covid k dead	0
2	unless biden plan call expedit approv rapid te...	1
3	bad resurrect floridian die covid irrespons ac...	0
4	christmasparti look realli upset mani peopl so...	0

Fig. 5. Representation of the data cleaning and preprocessing

- Removing special characters such as numbers, tabs, characters and make the Text have the same writing, in this case all are in lowercase this was done with the cleanString function.
- Elimination of empty words known as prepositions, pronouns, conjunctions, for the simple fact that do not provide any extra information, to achieve this we applied the function of stopwords taken by nltk library.
- The Stemming function was applied in order to have the text with the morphological normalization in a more accurate way compared to the lemmatization function.
- Tokenizer function was applied in order to turn the data into arrays.

In conclusion the two datasets were used only text preprocessing in particular normalization and cleaning and a little bit of transformation with the only objective that during the training of the models, the data have a good scope.

V. MODELLING

A. Test Design

The application of the models on the datasets were three, two of the models are related with the Text comment classification, while the third one model is based on the generation of Tweets, in which for two models it worked with known Framework Keras, which was created to perform deep learning neural networks, for that reason each of them can be build in a better way. So, the first model for the classification Text was Embedding Layers which can be used for neural networks on text data. Embedding, requires that the input data be integer encoded, so that each word is represented by a unique integer, also it can be used in a variety of ways like used alone to learn a word embedding that can be saved and used in another model later, used as part of a deep learning model where the embedding is learned along with the model itself and used to load a pre-trained word embedding model, a type of transfer learning??, and the second model as generator is also with the help of Keras but the model algorithm is called Char because it was as a sequential model constructed by a layers which is also called as Character-level text generation with LSTM.

B. Parameter Settings

The parameters used to score each model were Accuracy and a classification report in which the F1 Score, Recall and Precision metrics are found, and also for the generative model the Loss and Accuracy plots were tested.

C. Models

As an experiment, the Logistic Regression model was implemented as part of the Machine Learning models, although it gave us good results, it is not possible to use it since the objective of the models is to be neural networks, on the other hand, to obtain the Text Comment classification, the embedding layer was used, which some results.6 It can be

Model: "model_1"		
Layer (type)	Output Shape	Param #
input_2 (InputLayer)	[(None, 100)]	0
embedding_1 (Embedding)	(None, 100, 128)	2560000
bidirectional_1 (Bidirection	(None, 100, 100)	71600
global_max_pooling1d_1 (Glob	(None, 100)	0
dropout_2 (Dropout)	(None, 100)	0
dense_2 (Dense)	(None, 50)	5050
dropout_3 (Dropout)	(None, 50)	0
dense_3 (Dense)	(None, 1)	51
Total params: 2,636,701		
Trainable params: 2,636,701		
Non-trainable params: 0		

Fig. 6. Text Comment classification model

seen, that the model was made through several inputs, the first one was the initial layer which had a maximum length of 100, from there comes the second layer which is initialized with random weights and will learn an embedding for all words in the training dataset. words in the data, long term dependencies could also be called the Embedding layer, from there other layers were applied, until the dense layer was reached. On the other hand, we have the generative model which is more related to the sequential model. The model skeleton started with a single hidden layer LSTM with three parameters, then layers was dropout, LSTM and dropout, as get the output layer which is a dense layer that uses the SOFTMAX activation function as parameter to obtain a probability prediction for each the data registered in order to obtain a probability prediction for each of the data between 1 and 0,7 To conclude, it can be observed that each model was

Layer (type)	Output Shape	Param #
lstm (LSTM)	(None, 40, 128)	92672
dropout (Dropout)	(None, 40, 128)	0
lstm_1 (LSTM)	(None, 128)	131584
dropout_1 (Dropout)	(None, 128)	0
dense (Dense)	(None, 52)	6708
Total params: 230,964		
Trainable params: 230,964		
Non-trainable params: 0		

Fig. 7. Generator model

made in a different way, especially in the way in which the layers are constructed. It should also be noted that although each model was built for different functions, the best model

considered so far would be the Text classification model, since compared to the Logistic Regression model, both have a good accuracy, on the contrary that the generator model has a medium accuracy.

D. Model Description and Results

Obtained results of each are described. For model which is the classification model, obtained results are as follow:

Result Table				
id	Precision	Recall	f1-score	Support
0	0.5594	0.3335	0.4179	3121
1	0.5114	0.7264	0.6002	2997
Accuracy	0	0	0.5260	6118
macro avg	0.5354	0.5300	0.5091	6118
weighted avg	0.5359	0.5260	0.5072	6118

TABLE I
CLASSIFICATION METRICS RESULT

In table I are reported the metrics of the text classification, we got an accuracy of 0.5260 and a precision of 0.5594 for the positive comments and 0.5114 for negative comments. For the text classification we got in the restored model an accuracy of 94.71%, which is the neural network.

Also, the accuracy of logistic regression was 0.66 and the generator model got a 0.333, as we know there are no a good metrics to evaluate the model, but we consider that for this moment, the model with a best performances was the neural network Text Classification, compare with the logistic regression, also it could be said that the generator model is not bad but it has a standard accuracy for that reason needs more improvements, and also considered other metrics for example ROC-Curve, or Classification Report as we did in the model with the Text Classification.

VI. EVALUATION

The purpose of this stage of the methodology, an attempt is made to evaluate the presented models, however this evaluation is from business objectives point of view that is required in the data mining objectives.

A. Assessment w.r.t (Business Success Criteria)

From a business perspective, the ability to recognize, classify and generate text with an acceptable percentage of credibility has been established as the main success criterion. This criterion may be somewhat subjective, so it is inevitable to rely mainly on the success criterion. From the data mining results, it is more specific and accurate.

- Model for objective 1: This model works "good" because it can classify any comment, whether it is positive or negative. By exploring the results, we can say with certainty that this model is acceptable.
- Model for Objective 2: Specifically, for this model, we encountered some problems when trying to generate readable content, because the model reached the point where it stopped learning and was unable to generate any type of words. After many tests to find a good model above, we got an acceptable model that can create readable words.

VII. DEPLOYMENT

The objective of this phase of CRISP-DM methodology it is to explain to the final user how to put in operation the created project as well as to present the results in a easily understand.

A. Deployment Plan and Maintenance

The supervision and maintenance of the implementation of the project is an important stage, because the data used to execute the project is downloaded from Twitter on a specific topic, because the download Third-party information will bring certain formatting issues. The amount of data on the move is an important reason why it is necessary to carefully extract samples and always back up the data. Data mining should be carried out in a not too long period of time, because every day someone is posting to discuss the selected topic. The following process can be established as a supervision and maintenance plan:

- Extract and store data every two days, and save the obtained information in the database.
- Each result obtained should be presented graphically in order to better understand the value obtained.

B. Visualizations

In future works, a dashboard will be implemented. That because having only classified the tweets is not much relevance, if it cannot give concise information and that the user understand whats happens just by looking for a few seconds. The dashboard will show bar charts to see the classification between negative an positive tweets. Other kind of graphs will be a map of countries that will show the number of tweets about Covid-19 by each state, where the lighter colors mean that they made few twtwets, while darker colors will show that a large number of tweets were made about the Covid. The graphs will be programmed using Google sheets, Tableau and Python. The data will be provider in real time.

C. Final Report and Presentation

The report will be presented in a live presentation. The use of CRISP-DM methodology allowed to find out how text was classified and how the model generated acceptable results. For the initial data mining goals that have been set, two of them have been achieved, text generation and text classification. Review the different stages we followed to reach our goals Goal: The first stage is not as complicated as we expected, which is to extract data from Twitter and store it in a database. The extracted data is only filtered in English. Of course, the words tracked are related to the topic of coronavirus, because we need a large amount of data, and it takes time to download this information, because twitter has a time limit. On the other hand, the toxic comment database was retrieved from Kaggle. When we already have a database for data mining, we analyze the structure of the data and the information it contains. Next, all data goes through a pre-setting stage, and the data is cleaned, transformed, and vectorized so that it can be used in different models.

After that, a testing phase was carried out to understand which parameters could provide us with the best results. This stage took us a lot of time, because we did not have a powerful machine to run the models, each model took us about a day and a half to get the results, of course, the desired results did not happen until different tests were carried out. Once the experimental phase is completed, a model is created to train the final model and obtain the final result. As we mentioned before, the end result is "acceptable", and the plan for the future is to develop a dashboard in order to present the information obtained in visual way.

D. Experience Documentation

Documentation is a fundamental part of the project development process; improving our goals and every step we are completing helps us have a complete vision of the needs of the project in order to fully satisfy them, perhaps not in order but in order. This report is structured in accordance with each stage of CRISP-DM, which helps to explain in detail the content of execution and the results obtained, so that it can be presented in a more understandable way. The whole project is a challenge for us, it is difficult to try to keep the model and the various parts of the visualization together.

REFERENCES

- [1] "15 python libraries for data science you should know," Oct 2021.
- [2] T. C. Nokeri, *Big Data, Machine Learning, and Deep Learning Frameworks*, pp. 7–14. Berkeley, CA: Apress, 2022.
- [3] A. t. A. M. Koyejo, "Python framework for data science," Dec 2020.