WARSAW UNIVERSITY OF TECHNOLOGY
Faculty of Mathematics
and Information Science

Cloud Computing

Project of
Twitter Hate Speech Detection using AWS

Done by:

Amir Ali

M.Sc. Data Science

29 May 2022

Outline

1. Problem Statement

The term hate speech is understood as any type of verbal, written or behavioral communication that attacks or uses derogatory or discriminatory language against a person or group based on what they are, in other words, based on their religion, ethnicity, nationality, race, Colour, ancestry, sex or another identity factor. In this problem, we will take you through a hate speech detection model with Machine Learning and Python.

Hate Speech Detection is generally a task of sentiment classification. So, for training, a model that can classify hate speech from a certain piece of text can be achieved by training it on a data that is generally used to classify sentiments. So, for the task of hate speech detection model, we will use the Twitter tweets to identify tweets containing Hate speech.

2. Project Description:

A simple web base detection system built on AWS cloud, which detect the hate and free speech of given input text. This is based on classification machine learning model created using Amazon SageMaker. The definition and provision of the resources on AWS cloud is done through the AWS CloudFormation template.

Below you can see the interface:

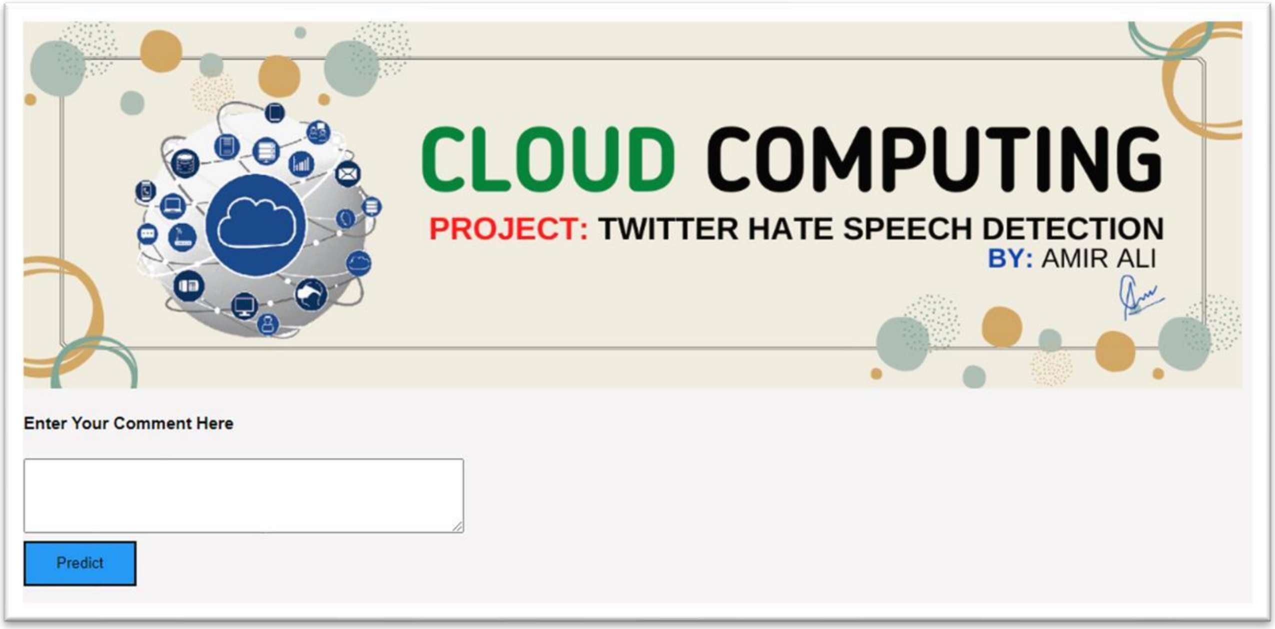


Figure 1: Hate Speech Detector System

3. Functional and Non-Functional Requirements

In this part I explain the functional and non-functional requirement of the project.

3.1 Functional Requirements

1. Scrapping twitter hate/Free Speech though API
2. Dumb data into S3 Bucket
3. Create Model on Sagemaker and connection through endpoint with Lambda Function
4. Maintain the Lambda Function between Web interface API and SageMaker Model Endpoint.
5. User interfaces get the output though Web Interface API.

3.2 Non-Functional Requirements

1. Ensure high availability of twitter hate/free speech data.
2. The request processed within 30 seconds.
3. The system is scalable and cost effectiveness.
4. The system is secure because there no info of user regarding registration.
5. Easily to distinguish the hate and free speech based on model accuracy.

4. Required Technologies and Libraries:

In this part, we write down the required Technologies and libraries that we used in this project

4.1 AWS Services

1. EC2
2. Lambda
3. S3 Bucket
4. SageMaker

4.2 Required Libraries

1. Boto3
2. Sagemaker
3. Numpy
4. Pandas
5. Matplotlib
6. Re
7. Nltk
8. Strings
9. Wordcloud
10. Sklearn
11. Tensorflow
12. Flask
13. Pickle

4.System Architecture:

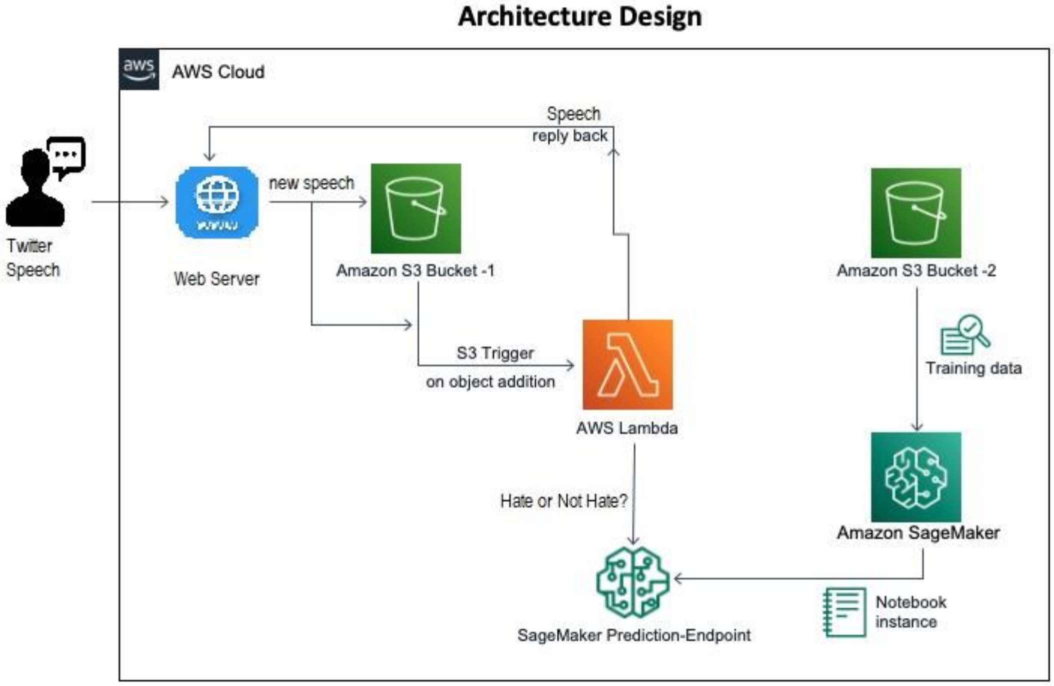


Figure 2: System Architecture

4.1 Standing up the AWS resources and Machine Learning hate speech detection using AWS Sagemaker

4..1.1 Overview

SageMaker is basically a fully managed service that provides the platform to build, train, and deploy ML models and it removes the heavy lifting from each step of the ML process to make it easier to develop high-quality models [1].

4.1.2 Data Information

Dataset using Twitter data, is was used to research hate-speech detection. The text is classified as: hate-speech, offensive language, and neither. Due to the nature of the study, it’s important to note that this dataset contains text that can be considered racist, sexist, homophobic, or generally offensive. [2]

Below you can see the World cloud of hate and free Speech that I build after data preprocessing (For Reference: See Source Code File)

4.1.3 Build and Train the Model

4.1.3.1 Create an Amazon S3 bucket

Use the Cloud Formation template (For Reference: See CloudFormationStack.json source file) to create the S3 buckets (for storing the twitter hate speech data), AWS Sagemaker endpoint, notebook instance, Lambda function and appropriate roles.

AWS CloudFormation:

- Create a CloudFormation template to represent all the infrastructure resources (ex. Lambda, S3 bucket, etc.) and permissions (IAM policies, roles, etc.).
- The template take the prediction endpoint.

4.1.3.2 Create a managed Jupyter Notebook instance

In this part we use an Amazon SageMaker managed Jupyter notebook instance to prepare and process data and write the code to train the model. An Amazon SageMaker notebook instances is a fully managed ML compute instance running the Jupyter Notebook application. Amazon SageMaker manages creating the instance and related resources.

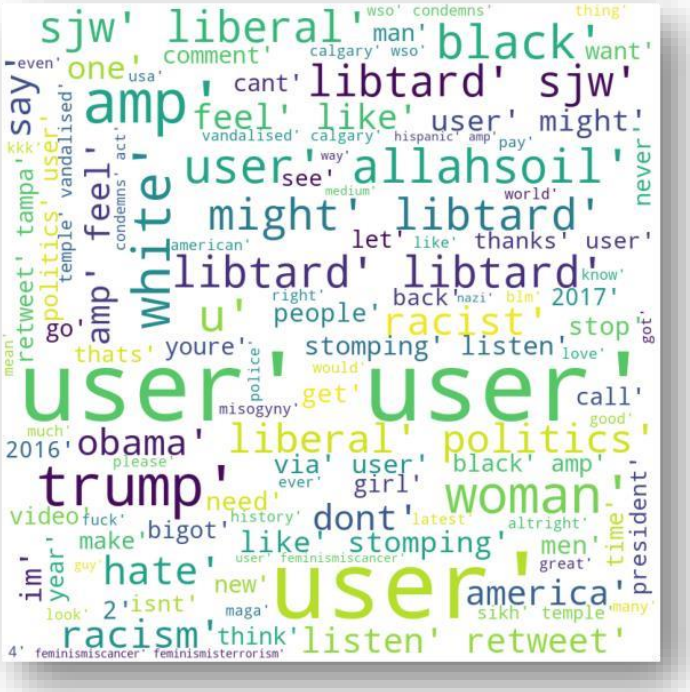


Figure 3: Hate Speech Wordcloud

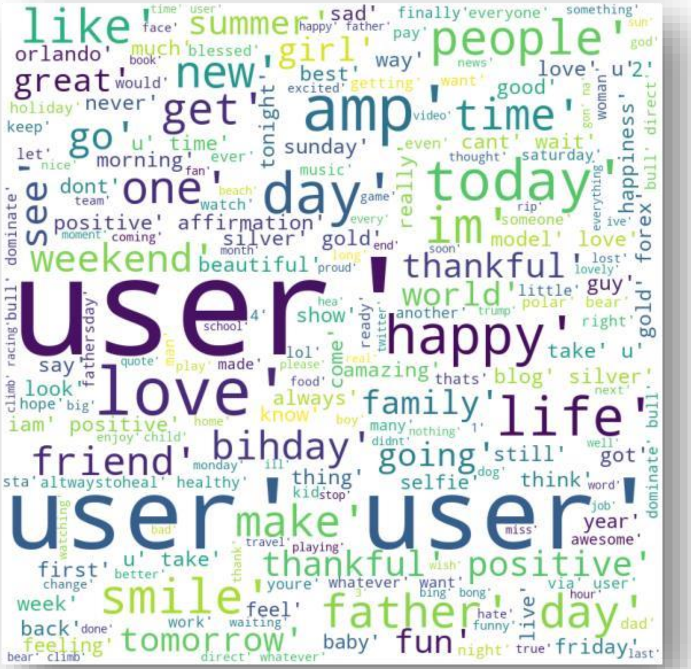


Figure 4: Free Speech Wordcloud

4.1.3.3 Write a Python Code to Build, Train and predict (Reference: see source Code File

- **Data Preprocessing:**

In this part, we did data preprocessing. firstly, we did text cleaning with the help of Regular Expression and then we apply preprocessing 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. After that, we apply the IF-IDF model for feature extraction, and we took 200 most frequent words from dictionaries. In the end, we split our data for training and testing. And ratio for training and testing is 0.8 and 0.2 respectively.

- **Build the Support vector Machine Classifier Model**

In this part, we build our model to predict hate and free speech. we implement Support Vector Machine using Scikit-Learn.

In Support Vector Machine we separate a data point into class attribute using hyperplane to separate our data. In this technique, we lot each data item as a point in n-dimensional space (where n is the number of features you have) with the value of a particular coordinate. Then we perform classification by finding the hyperplane that differentiates the two classes very well. [3]

- **Result Evaluation and make a new Prediction**

In this part, we evaluate the result of our model. First, we visualize the confusion matrix and see the incorrect predictand and then we calculate the following score ("Accuracy", "Precision", "TPR", "FPR", "F-Score", "Specificity", "Error" and "Roc Area") and visualize them.

In the end based on our model accuracy we make a final prediction on new example data.

4.2 Setting up Lambda Function

In this step we are setting up Lambda Function

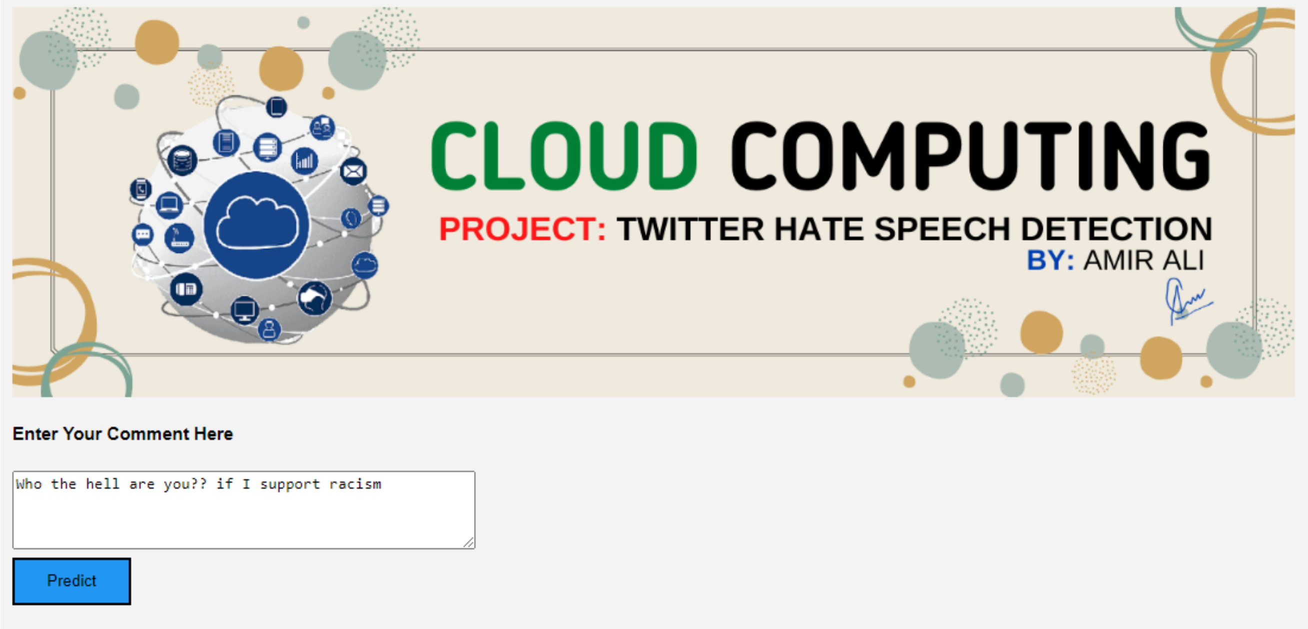
- Navigate to the lambda function created as part of the cloud formation template. From there, set up the trigger for the Lambda as y S3 bucket and we make sure that S3 has required IAM permission for the Lambda function resource.

- For the code part of the Lambda function, use the Python code. If necessary, add layers in the lambda function for numpy and other libraries. Make sure that Lambda has required IAM permissions for the S3 resource, Web Server etc.

4.3 Setting up the Simple Website for the input and output of our application

The frontend application created using the Flask Framework. In the application user gave Hate/Free Speech text as an input of and on the other hand access the prediction result as output. The client-side application will communicate with the AWS Lambda service via REST API. After sending a request to the AWS Lambda service, the valuation of the input(text) from the model will be returned in the response.

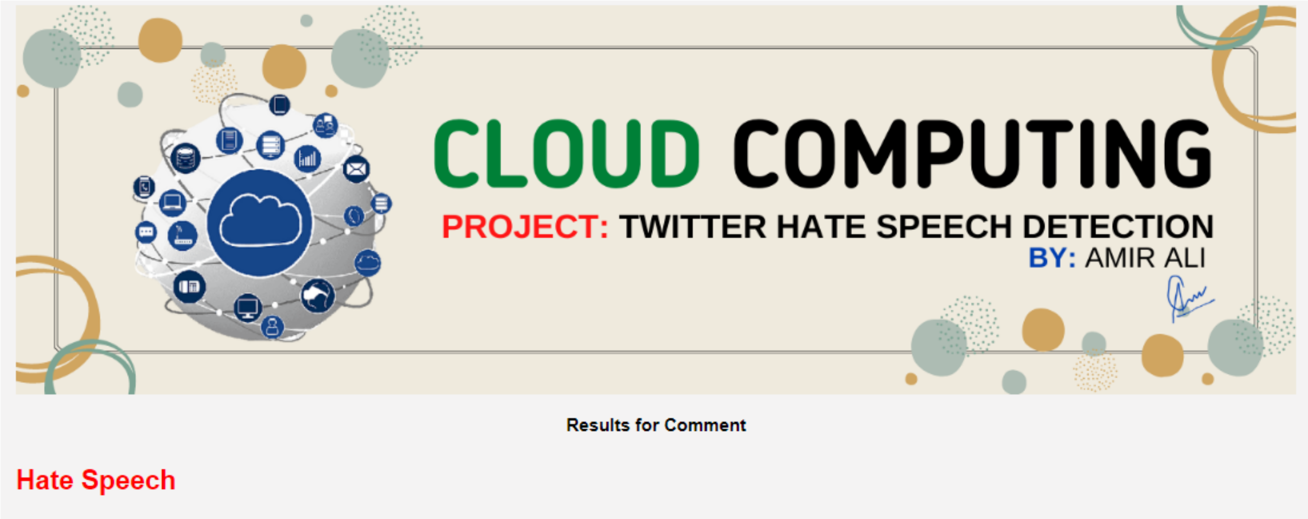
Below you can see the application working. In Text box we gave the input and after click on the blue button it will predict our input and then base on our input it return the output either is Hate speech or Free Speech.



Enter Your Comment Here

Who the hell are you?? if I support racism

Figure 5: Input Example



Results for Comment

Hate Speech

Figure 6: Output Example

References:

[1] https://aws.amazon.com/sagemaker/mlops/

[2] https://www.kaggle.com/datasets/mrmorj/hate-speech-and-offensive-language-dataset

[3] https://medium.com/machine-learning-researcher/support-vector-machine-a57e575b05bb