

DOI: [http://dx.doi.org/10.21123/bsj.2021.18.2\(Suppl.\).0937](http://dx.doi.org/10.21123/bsj.2021.18.2(Suppl.).0937)

A Word Cloud Model based on Hate Speech in an Online Social Media Environment

*Valentina Ibrahim**

Juhaida Abu Bakar
Alaa Fareed Abdulateef

Nor Hazlyna Harun

School of Computing, College of Arts and Sciences, Universiti Utara Malaysia, Sintok, Kedah

*Corresponding author: valentina_ibrahim@ahsgs.uum.edu.my, juhaida.ab@uum.edu.my, hazlyna@uum.edu.my, alaafareed85@gmail.com

ORCID ID: <https://orcid.org/0000-0002-0533-2456>, <https://orcid.org/0000-0002-2804-2160>, <https://orcid.org/0000-0002-6817-9874>, <https://orcid.org/0000-0001-5978-6077>

Received 28/3/2021, Accepted 13/4/2021, Published 20/6/2021



This work is licensed under a [Creative Commons Attribution 4.0 International License](https://creativecommons.org/licenses/by/4.0/).

Abstract:

Social media is known as detectors platform that are used to measure the activities of the users in the real world. However, the huge and unfiltered feed of messages posted on social media trigger social warnings, particularly when these messages contain hate speech towards specific individual or community. The negative effect of these messages on individuals or the society at large is of great concern to governments and non-governmental organizations. Word clouds provide a simple and efficient means of visually transferring the most common words from text documents. This research aims to develop a word cloud model based on hateful words on online social media environment such as Google News. Several steps are involved including data acquisition and pre-processing, feature extraction, model development, visualization and viewing of word cloud model result. The results present an image in a series of text describing the top words. This model can be considered as a simple way to exchange high-level information without overloading the user's details.

Key words: Hate crime, Python, Social network, Word cloud.

Introduction:

Natural Language Processing (NLP) is a discipline within AI which focusses on allowing computers to understand a process human language. Technically, NLP's primary task is to program computers to analyze and process a huge amount of natural language knowledge (1,2). It is a technology used every day by many people and has existed for many years. NLP is drawn from many disciplines, including computer science and computer linguistics, in its effort to fill the gap between human communication and machine understanding (3). This field is mainly concerned with the basic yet efficient relationship between humans and computers (4). The word 'cloud' or 'tag clouds' is a graphical representation of textual data terms (5).

The word 'cloud' is a visualization which abstracts a document by providing an overview of the most commonly used terms. Visual text abstractions can provide users with valuable insight and help them understand the information in a document without reading the full text (3). The

most common words are broad and the less famous words are smaller in number (6). Google indicates that a 'word cloud' is a picture consisting of words used in a particular text or subject, in which each word's size indicates its frequency or significance (7). The more often a specific word appears in the text, the bigger and bolder it appears in the cloud of words. It is important because it reveals the key; on the surface, pop brand names and keywords float.

The word cloud is also quick to go over the text to create patterns instead of analysis that takes time. It is used for the study of various forms of news and social media texts. Hate speech is of broad and current importance in the world of social media at present (6). The alias and resilience afforded by the internet have facilitated the way for users to communicate offensively. In addition, tools can be used to automatically detect hate speech, as the amount of online hate speech increases (6).

There are many definitions for hate speech, among them, "the language that attacks or

diminishes, that motivate violence or hate against groups, based on specific characteristics such as religion, physical appearance, descent, national or ethnic origin, sexual orientation, gender identity or other, and it can occur with different linguistic styles, even in delicate forms or when humor is used" (8). Pertaining to hate speech, hate crimes are a "type of violation of the law whose primary motivation is the existence of preconception regarding the victims" (9). The victims usually belong to a certain group defined basically by previous mentioned attributes. Usually, hate crimes are invoked by singular widely broadcasted events (terrorist attacks, riots, uncontrolled migration, demonstrations, etc.) (9). These incidents typically serve as causes, and their influence on social media significantly increases, making social media a tracker in the real world and a source of valuable crime prediction knowledge. Social networks are loaded with posts from people who advocate punishment against various targeted groups. If these messages are gathered over a period of time following an outrageous incident, they can be used in all stages to examine hate crimes, which include climbing, stabilization, duration, and decline of the threat. Therefore, social media observation is a priority for forecasting, identification, and review of hate crimes (9).

The growing existence of social media has caused a plurality of possibilities. However, it also has created problems of a completely new dimension, particularly through the massive number of users. It is rather difficult to control online content. In the last couple of years, machine learning and natural language processing approaches have been considered to detect harmful user content on the web (10). Many countries have introduced laws to oppose hate speech. Beyond that, online platforms apply individual language regulation policies built upon their definitions, thus, the need to find a way to detect hateful news and to demonstrate it quickly using known language like Python. This method can be propagated to reduce the hateful speech over the network media. This paper is aimed at creating a word cloud that shows clearly the selected parts of the hate speech text that most likely contain hate crime news. Others have used many different algorithms to detect hateful words in social media, but this study uses Python because it has ready libraries, and is easier to change its parameters.

There are different methods to detect hate words in the network, but the goal of this paper is to use word cloud as a tool to analyze huge datasets for Google News to detect hate crime news. Python generates the word cloud using a dataset that will be

analyzed. This study chooses a google news dataset taken from the ProPublica data store; it has a free Documenting Hate News Index (Raw Data), and also to download the necessary libraries that will be used when programming (11).

The dataset is extracted from a site that includes a set of news stories about hate crimes collected by Google News, which include the title, date, publisher, location, keywords, and a summary of each article. The data were compiled from 10 March 2020 to 16 March 2020. The link was retrieved from ProPublica (11). This paper deals only with hate crime news, written only in the English language.

Next, Section 2 presents the related studies on different techniques used in classifying and detecting hate speech and abusive activity via social media. This section also describes the importance of adapting text mining techniques to identify hate speech. Section 3 states the methodology of the research. Section 4 presents the proposed word cloud technique. This section includes the experimental analysis and discussion. Section 5 describes the discussion and evaluation. Finally, this paper concludes with section 6 with the feasible future scopes.

Related Studies:

The definition of hate speech has become common in recent years particularly with the extend of the internet; many works have been published in regards to the detection, awareness and explanation of hate speech and its process (12). However, only a few datasets have been compiled, clarified, and published by other researchers pertaining to abusive activity on social media. Indeed, there is a general lack of methodical hate content control, reporting and data collection, which restricts of the findings of the most recent studies. Regardless of this limitation, studies on hate speech detection computational methods have been growing, concentrating primarily on adapting text mining techniques to the problem of automated hate speech detection. The most common approach found when analyzing the methods for detecting hate speech in social media is to create machine learning models for the classification of hate speech.

Most studies seek to follow techniques that are already known in text mining for the problem of automated detection of hate speech, such as the use of dictionaries and lexicons. The study by (13) aimed to set lexical baselines for classification by applying direct classification methods using a dataset illustrated for this reason. This study achieved 78 percent accuracy in the identification of posts across three classes (Hate, Offensive, and Ok)

using an N-Gram and linear Support Vector Machine (SVM) approach to multi-class classification results. This led to the principle of preprocessing a text using machine learning prior to classification.

Yunhai Wang et al., (14) introduced EdWordle, which enables a neighborhood preserving editing method to preserve words at predictable and near locations during and after the editing process. This approach allows users to transfer and modify words while maintaining the neighborhoods of other words by combining a constrained rigid body simulation with a neighborhood-aware local Wordle algorithm to update the cloud and construct very compact layouts. EdWordle allows users to create new types of word clouds in which the location of words is carefully edited, such as storytelling clouds. Semantic word clouds are increasingly difficult to edit without losing their semantic layout. This void is being filled by EdWordle. However, this approach is occasionally used as a data analysis tool with the need to accurately represent the underlying data.

Jin, (2015) on the other hand, focused on creating a graphical user interface (GUI) program to create word cloud maps with simple operations (15). The program was built on Python; the algorithm is based on simple linear, power, and logarithmic font size representation, providing users with text mining tasks with a versatile, customizable, and user-friendly method. The platform has been shown to support mixed multi-language environments, offering a broader variety of applications.

Felix et al. conducted several user studies aimed at exploring the graphic design space of word clouds (16). The research focuses on and involves many variations of spatial layout and meaning encoding. The study discovered that success depends on the task used and believe that improved output can also signal an excessive effect of visual encoding on attention when searching for a high frequency word. Martins et al. (12) introduced a mixture of lexicon-based and machine learning methods used by an emotional approach by sentiment analysis to predict hate speech found in a text. Using a dataset annotated for this purpose, the study applies classification methods. The

framework of this study uses Natural Language Processing (NLP) techniques (such as Naive Bayes, Random Forest, and SVM) as features to extend the original emotional knowledge dataset and provide it for the classification of machine learning. Using the emotional data found in the text helps improve the precision of the identification of hate speech. In hate speech recognition, this study obtained results of 80.56 percent accuracy.

Mossie and Wang (17) built a model based on Apache Spark to identify posts and comments on Amharic Facebook into hate and not hate. Random Forest and Naïve Bayes were used by the authors for learning, and Word2Vec and TF-IDF for function selection. The model based on Word2Vec embedding performed best with 79.83 percent precision, checked by 10-fold cross-validation. Cross-validation is a technique for evaluation a prediction model by splitting the original sample into a training set to train the model, and a test set to evaluate it (18). The proposed approach achieved a promising outcome with a special spark function for big data. They developed an Amharic text hate speech identification model that analyzes posts and comments to use spark machine learning techniques to recognize hate speech.

Salminen et al. (19), on the other hand, found that there is a shortage of models using multi-platform data for online hate detection. Therefore, with those criteria, they received a total of 197,566 comments on four platforms, which are YouTube, Reddit, Wikipedia, and Twitter. This study then experimented with multiple classification algorithms and feature representations (Logistic Regression, Naïve Bayes, Support Vector Machines, XGBoost, and Neural Networks) with Bag-of-Words, TF-IDF, Word2Vec, BERT, and their combination. While all the models outperformed the keyword-based baseline classifier significantly, XGBoost performed the best (F1 = 0.92) using all features. Analysis of feature significance shows that BERT features are the most powerful for the detection.

Methodology:

The research approach consists of two stages, namely data collection and analysis, as well as the creation of model. Figure 1 shows the research structure.

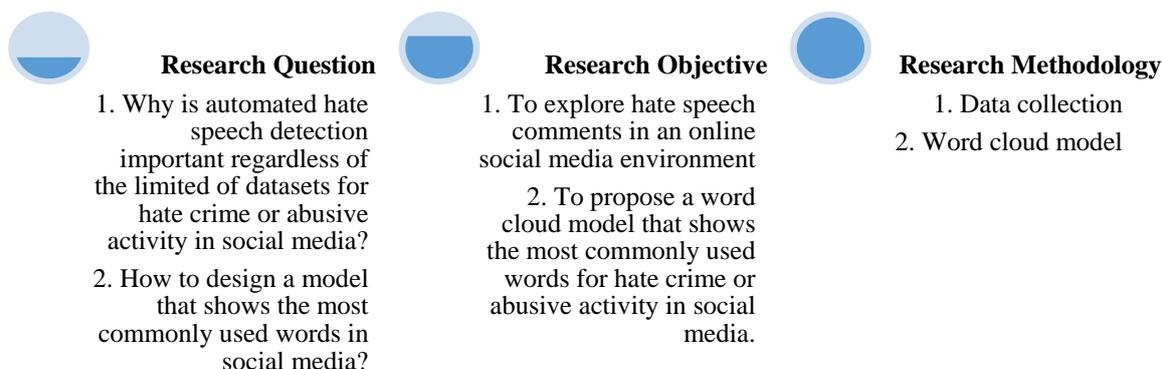


Figure 1. Research Structure

Phase 1: Data collection and analysis

RO1: To explore hate speech comment in an online social media environment.

To address the objective, a data collection was performed from several sources. The collection of data helps a person or organization to answer specific questions, analyze results and make predictions about potential probabilities and trends. For preserving the credibility of science, making informed business decisions, and ensuring quality assurance, and accurate data collection are important.

The used dataset was downloaded from a website that collects media stories about hate crimes and cases of racism, which are compiled by Google News. The "keywords" column includes the names and locations found in the "articles" column in the news reports. It was launched on 18 August 2017, by Pitch Interactive. The data included the title, date, publisher, location, keywords, and a summary of each article from ProPublica (11). The date of the data was for the duration from 10 March 2020 to 16 March 2020.

The outcome of Phase 1 is the collection of hate speech dataset. It will serve as the input for Phase 2. With the identification of the hate speech datasets, the first objective of the research (RO1) is achieved.

Phase 2: Model development

RO2: To propose a word cloud model based on hate speech in an online social media environment.

Word clouds (also known as a tag cloud) are recognized as an image consisting of words used in a specific text or topic in which their frequency or meaning is indicated by the size of each word. This seemingly simple way of presenting data is extremely flexible. In this phase, several steps are involved including import the libraries to use Python, read the CSV extension data file, iterate through the data file to convert it to a long text, put the mask and evaluate stop words, create the word cloud, and then view the word cloud. The result would present an image in a series of text

describing the top words and the significant of the word cloud in the social media online environment can be seen.

There are five work processes for this modelling, which are (1) data acquisition and pre-processing, (2) feature extraction, (3) model development, (4) visualization, and (5) result analysis. The inputs of the processes are the variables obtained from the data collection in Phase 1, and the outputs of the processes are the generated word cloud.

- (1) Data acquisition and pre-processing – Data is collected from the open-source dataset for further pre-processing. To run code in Python, several libraries have been identified to be imported namely Numpy, Pandas, Matplotlib, Pillow, and Wordcloud. In this work, Comma delimited (.csv) file is read and stored as Pandas DataFrame.
- (2) Feature extraction – Informative variables are identified in this phase. Two necessary strings namely 'comment words' and 'stop words' have been initialized to generate the word cloud.
- (3) Model development – In this stage, the process of converting each row of the 'Article Title' column to a very long, single line of text is generated. PorterStemmer function and WordNetLemmatizer function have been designed to develop the word cloud. The purpose of PorterStemmer function is to normalize term. To ensure homogeneity of all the words, lowercase function is used. WordNetLemmatizer function is used to group together various types of inflected word. Then, all the words have been stored together to generate word cloud. After that, the mask of circle shape for word cloud must be generated. This process is called putting the mask. Then, this study chose the stop words to prevent them from appearing in the word cloud.

- (4) Visualization - Once the process has been constructed, a word cloud visualization is developed to represent the data.
- (5) Result – Finally, the readers would notice the larger, bold words and understand their importance in the text. Through word cloud, this study is able to create a picture that emphasizes on the main hateful words in one news in a newspaper.

The outcome of Phase 2 is the proposed model. With the outcome of the proposed model, the second objective of the research (RO2) is achieved.

Results and Findings

Word cloud or tag cloud is a cloud filled with several different word sizes that represent the frequency or significance of each word (20). The module of word cloud has the advantages of it fills all available space, it can use arbitrary masks, it has a simple yet efficient algorithm that can be easily modified using Python (21).

Seven essential steps are described in this section as follows, (1) Importing the libraries, (2) Reading CSV file, (3) Setting the comment and stop words, (4) Iterating through the data file, (5) Putting the mask and determining stop words, (6) Creating the word cloud and (7) Displaying the word cloud.

Step 1: Importing the Libraries

The first step is importing the libraries. Numpy, Pandas, Matplotlib, Pillow, and Wordcloud are several packages that need to be installed. One of the most common and helpful libraries used to handle multi-dimensional arrays and matrices is the Numpy library. It is also used to conduct data analysis with the Panda library. The Python OS module is a library that is built-in. Matplotlib is a simple visualization library that allows many other libraries, including seaborn or word cloud, to run and plot on its foundation. A kit that allows picture reading is the pillow library. It is a PIL - Python Imaging Library wrapper. To read in image as the mask for the word cloud, this library is important. First is to load all the libraries required as in Figure 2.

```
import numpy as np
import pandas as pd
from os import path
from PIL import Image
from wordcloud import WordCloud, STOPWORDS, ImageColorGenerator
import matplotlib.pyplot as plt
% matplotlib inline
```

Figure 2. Importing Libraries

ImageColorGenerator is a base class to recolor the words in the word-cloud image.

Step 2: Reading CSV File

In the next step, CSV file is read and stored into a Pandas DataFrame as illustrated in Figure 3. The Pandas DataFrame are always easier and faster to use when working with large datasets. The column required for the word cloud generation of this study can be easily accessed from the Pandas DataFrame. The result snippet in the form of the Pandas DataFrame is illustrated in Figure 4.

```
df=pd.read_csv('20170816_Documenting_Hate.csv',encoding='latin
-1')
df['Article Title'].head(100)
```

Figure 3. Reading Data File

```
0 Pizza Hut driver who killed co-worker with sho...
1 Residents of NJ township receiving KKK promoti...
2 House OKs bill to expand Kentucky's hate crime...
3 Amid Protests, 'Blue Lives Matter' Bill Passes...
4 Lafourche inmates charged with hate crimes in ...
...
95 Suspect In Kansas Shooting Faces Possible Hate...
96 Feds Probe Kansas Shooting As Possible Hate Crime
97 Kansas shooting raises safety concerns for for...
98 Idaho locker room attacker gets probation afte...
99 Cuomo unveils hate crime text line
Name: Article Title, Length: 100, dtype: object
```

Figure 4. Result snippet that provides Pandas DataFrame

Step 3: Setting the comment and stop words

In this step, two necessary strings have been initialized to generate the word cloud. The 'comment_words' is the string that will be used to store all the words of the 'Article Title' column in a single line of text. The 'stop_words' are used to store all the words that are very commonly used in the English language, such as 'the', 'a', 'an', 'in'. These words will be later filtered while generating the word cloud. Figure 5 shows the process.

```
0 Pizza Hut driver who killed co-worker with sho...
1 Residents of NJ township receiving KKK promoti...
2 House OKs bill to expand Kentucky's hate crime...
3 Amid Protests, 'Blue Lives Matter' Bill Passes...
4 Lafourche inmates charged with hate crimes in ...
...
95 Suspect In Kansas Shooting Faces Possible Hate...
96 Feds Probe Kansas Shooting As Possible Hate Crime
97 Kansas shooting raises safety concerns for for...
98 Idaho locker room attacker gets probation afte...
99 Cuomo unveils hate crime text line
Name: Article Title, Length: 100, dtype: object
```

Figure 5. Removing punctuations

```
punctuations = '!()-[]{}:;'\",<>./?@#%$^&*~'
no_punct = ""
for char in df['Article Title']:
    if char not in punctuations: no_punct = no_punct + char

#display the unpunctuated string
print(no_punct)
import nltk
nltk.download('wordnet')
```

Step 4: Iterating through the data file

After data file have been stored into a Pandas DataFrame, each row of the 'Article Title' column is converted to a single line of text. The first step is to use PorterStemmer to eliminate the commoner morphological and inflectional endings from words(). Its main use is as part of the process of normalization of a term. After that, and for the homogeneity of all the words, all the words are converted to lowercase using the .lower() function.

Then WordNetLemmatizer() is used, which is the method of grouping together the various types of a word that are inflected in order to evaluate them as a single object. Lemmatization is comparable to stemming, but it provides the words with context. Therefore, it links words with similar meanings to one word. Finally, all the words are joined and stored to the variable 'Article Title_words' using the function .join(). The variable 'Article Title_words' now contains all the words in a single long text necessary to generate the word cloud. The process is shown in Figures 6-7. Figure 8 shows a result snippet that includes iterate data file.

```
from nltk.stem import PorterStemmer
from nltk.tokenize import word_tokenize
from nltk.stem.wordnet import WordNetLemmatizer
from nltk.corpus import stopwords
from nltk.stem.porter import PorterStemmer
from nltk.tokenize import RegexpTokenizer import re
import nltk

sent=''
```

Figure 6. Importing stemmer and lemmatizer libraries

```
ps = PorterStemmer()
for w in no_punct:
    sent=sent + ps.stem(w)
corpus=[]
sent = sent.lower()
sent=re.sub("<?/?.*?>"," " &lt;?/?> ",sent)

sent=re.sub("(\\d|\\W)+"," ",sent)
lem = WordNetLemmatizer()
sent = [lem.lemmatize(word) for word in sent]
sent = "".join(sent)
corpus.append(sent)
corpus
```

Figure 7. Iterate data file process

```
[Pizza Hut driver who killed co-worker with shotgun found mentally competent
Residents of NJ township receiving KKK promotional fliers House OKs bill to
expand Kentucky's hate crimes law Amid Protests, 'Blue Lives Matter' Bill
Passes Kentucky House Lafourche inmates charged with hate crimes in jail
attack Police: Davis hate crime suspect had 'aspirations' to kill Fowler's
'Blue Lives Matter' bill is great tribute to our protectors, but...Arrest
made after bacon strips left at California mosque Arrest made after bacon
strips left at California mosque - KSBY.com | San Luis Obispo and Santa
Barbara Area News Report: San Diego violent crime rates lowest in decades
Blue Lives Matter Bills Would Make Targeting First Responders a Hate Crime
Reports Of 'Hate Crime' Are Exaggerated By Those Who Oppose Brex it New
bias crime dashboard shows increase in reports of Seattle hate violence
Tulsa man charged in neighbor's killing to plead in sanity SPLC says number
of anti-Muslim hate groups on the rise Watchdog: Number of anti-Muslim hate
groups on the rise Police investigate anti-Muslim flier at Louisiana mosque
Details Paint a Grim Picture in the Hate Crime Case Woman Facing Hate Crime
Charges In Attack On Davis Mosque Authorities: Davis Mosque Vandalism
Suspect 'Immediate Danger' to Public FBI Investigating Hate Crime at UMW
Teach NYS Calls for Increased Security Funding in Wake of Rising Hate Crimes
Senate amendment would add LGBT to state hate crime law Hate Crime Suspect
Arrested After Vandalizing a Mosque With Bacon First-Ever Tracker Of Hate
Crimes Against Asian-Americans Launched Why Does the Number of US Hate
Groups Keep Growing? Canadian Anti-Muslim Protesters Could Face Hate Crime
Charges Documents show physical evidence behind hate crime hoax arrest
Vandals damage dozens of headstones at historic Jewish cemetery Vandals
```

Figure 8. Snippet result of iterate data file

Step 5: Putting the mask and determining stop words The NikhilkumarSingh (22) study refers to build the circle shape mask in the word cloud. To prevent it from appearing in the word cloud, this study determines stop words. Not all mask images have the same format, resulting in different results, so with the WordCloud

function, it does not function properly. To verify that the mask is working, this study checks the Numpy array form. When the masking functions loaded with zero, change the masking functions to 255. The way the masking functions operate is that 255 should be the all-white portion of the mask rather than 0 (integer type). This value is

were bigger in size in regards to other words. Using word cloud to evaluate the top words in a text set, this study may infer the effectiveness.

Discussion and Evaluation:

Before using any method, the first thing to do is to review the function docstring and see all necessary and optional arguments using ?function. The parameters are font_path, width, height, prefer_horizontal, mask, contour_width, counter_color, scale, min_font_size, font_step, max_words, stopwords, background_color, max_font_size, mode, relative_scaling, color_func, regexp, collocations, colormap, normalize_plurals, and the attributes are ``words`` and ``layout``. Text is the only argument needed for a WordCloud object, while all others are optional. To use Python, it takes several steps to import the libraries, read the data file in CSV extension, iterate through the data file to convert it to one long text, put the mask and determine stop words, create the word cloud, and then finally display the word cloud. The result will show a picture identifying the top words in a collection of text. This tool can be considered a simple way to exchange high-level information without overloading the user's details.

There are many ways in which it is able to evaluate and develop the word cloud. Components that have been used as word cloud evaluation, such as common word technique, lemmatization or stemming, text cleaning, word cloud features, and further analysis of text.

- (1) Common word: In all the character scripts, there are repeated words which are familiar. The word clouds are currently created based on the word frequency, but an alternative would be TFIDF, which weighs the words based on the document's term frequency and the corpus frequency. Alternatively, to exclude other frequent words, a custom stop word list could be created.
- (2) Lemmatization/stemming: In this research, lemmatization and stemming have been applied. The first step is to extract the commoner morphological and inflectional endings from words by using PorterStemmer(). Its primary usage is as part of a term's normalization process. After that, and for the homogeneity of all the words, the .lower() function converts all the words to lowercase. WordNetLemmatizer() is then used, which is the method of grouping together the different forms of a word that are inflected as a single object to test them. Lemmatization is similar to stemming, but gives meaning to the words. It also binds words to one word with similar

meanings. It means that words that come from the same root word are related and do not occur in the word cloud many times, e.g., charge, charged.

- (3) Text cleaning: Further cleaning or data preparation measures are available that could be used. Several text cleanings have been implemented in this study, such as removing punctuations and deleting stop words.
- (4) Word cloud features: In this study, the white color and red background functions were checked with various word colors, background colors and word cloud parameters, but some of the parameters could be further optimized.
- (5) Further text analysis: From this dataset, several insights can be drawn. In this study, two strings were used, namely 'comment_words' and 'stop_words'.

Conclusion and Future Works:

The purpose of this paper is to define the term most used in hate crime news and visualize it using Python's word cloud. First, it was important to know what hate speech is and where it is mostly used. Hate speech has many meanings, many of which come from various sources. An easy way to locate hateful words easily and rapidly is to create a word cloud from Google News to detect hateful speech. This research provided a dataset from a website that contains a collection of news articles gathered by Google News about hate crimes, including the title, date, publisher, location, keywords, and a summary of each article. As it is a simple platform to apply the word cloud, this study used Python. The result is very satisfactory, and to get different shapes and colors, the context can be modified. Once users are able to overcome the misconception that word clouds are just beautiful and enjoyable objects, they can realize that these graphical representations can be practical and useful as testing and assessment tools. This article only deals with hate crime reporting, published in the English language. A simple option to locate hateful words easily and rapidly is to create a word cloud from Google News to detect hateful speech.

In terms of future work, the study should focus on identifying ways to make the word cloud part of any news published online and use it to detect unpleasant events in the next period, based on the results of the current word cloud. This can be extended to other social media platforms as well. Since word cloud is a fast way to instantly understand the most common crimes in some regions, states, or countries, strict future procedures

will therefore be taken to monitor these geographical areas, especially in polarized countries. This research can be expanded to include other languages, besides the English language. It will help to create a more positive society, ensuring that social media is widely used for the benefit of humanity. The findings can be used by social media sites to monitor user behavior and to commit hate crimes against specific individuals. This implies that users should explicitly report content to the provider of the site so that it can be evaluated and removed.

Acknowledgment:

This work is supported by the Malaysian Ministry of Higher Education and Universiti Utara Malaysia under SO Code 14442.

Authors' declaration:

- Conflicts of Interest: None.
- We hereby confirm that all the Figures and Tables in the manuscript are ours. Besides, the Figures and images, which are not ours, have been given the permission for re-publication attached with the manuscript.
- Ethical Clearance: The project was approved by the local ethical committee in Universiti Utara Malaysia.

References:

1. Jurafsky D. *Speech & language processing*. Pearson Education India. 2000.
2. Jurafsky D, Manning C. *Natural language processing*. Instructor. 2012;212(998),3482.
3. John M, Marbach E, Lohmann S, Heimerl F, Ertl T. MultiCloud: Interactive word cloud visualization for multiple texts. *Proceeding of Graphical Interface*. 2018;25–32.
4. Jain A, Kulkarni G, Shah V. *Natural language processing*. *Int. J. Comput. Sci. Eng.* 2018;6(1).
5. Milum J. *SAS® Visual Analytics: Text Analytics Using Word Clouds*. Atlanta, GA, USA. 2018
6. KopparthiHarika M, Mounika IC, Anuradha T, Sharon P. Hate Speech Detection in Tweets using Machine Learning Algorithm. *International Journal of Engineering Applied Sciences and Technology*. 2020;4(12),558-561.
7. Lepki L. How Do You Create a Word Cloud? *ProWriting Aid*. 2020. <https://prowritingaid.com/art/425/What-the-Heck-is-a-Word-Cloud-and-Why-Would-I-UseOne.aspx#:~:text=Google%20says%20a%20word%20cloud, appears%20in%20your%20word%20cloud>
8. Davidson T, Warmsley D, Macy M, Weber I. Automated hate speech detection and the problem of offensive language. In *Proceedings of the International AAAI Conference on Web and Social Media*. 2017;11(1).
9. Pereira-Kohatsu JC, Quijano-Sánchez L, Liberatore F, Camacho-Collados M. Detecting and monitoring hate speech in Twitter. *Sensors*. 2019;19(21),4654.
10. Dohmen C. *Detecting Hate Speech in Social Media- A Machine Learning Approach*. [Doctoral dissertation]:Hochschule für angewandte Wissenschaften Hamburg. 2019.
11. ProPublica. Retrieved from <https://www.propublica.org/datastore/dataset/documenting-hate-news-index>. 2020.
12. Martins R, Gomes M, Almeida JJ, Novais P, & Henriques P. Hate speech classification in social media using emotional analysis. In *2018 7th Brazilian Conference on Intelligent Systems (BRACIS)*. IEEE. 2018;61- 66.
13. Malmasi S, & Zampieri M. Detecting hate speech in social media. 2017. arXiv preprint arXiv:1712.06427.
14. Wang Y, Chu X, Bao C, Zhu L, Deussen O, Chen B, & Sedlmair M. Edwordle: Consistency-preserving word cloud editing. *IEEE transactions on visualization and computer graphics*. 2017;24(1),647-656.
15. Jin Y. Development of word cloud generator software based on Python. *Procedia Engineering*. 2017;174,788-792.
16. Felix C, Franconeri S, Bertini E. Taking word clouds apart: An empirical investigation of the design space for keyword summaries. *IEEE transactions on visualization and computer graphics*. 2017;24(1),657-666.
17. Mossie Z, Wang JH. Social network hate speech detection for Amharic language. *Computer Science & Information Technology*. 2018;41-55.
18. Vanschoren J, Van Rijn JN, Bischl B, Torgo L. OpenML: networked science in machine learning. *ACM SIGKDD Explorations Newsletter*. 2014;15(2),49-60.
19. Salminen J, Hopf M, Chowdhury SA, Jung SG, Almerakhi H, Jansen BJ. Developing an online hate classifier for multiple social media platforms. *Human-centric Computing and Information Sciences*. 2020;10(1),1.
20. Vu D. *Generating WordClouds in Python*. Datacamp community. 2019. Retrieved from: <https://www.datacamp.com/community/tutorials/wordcloud-python>
21. Foong NW. *Create Custom Word Clouds in Python*. Better programming. 2020. Retrieved from <https://medium.com/better-programming/create-custom-word-clouds-in-python-841563933e73>
22. NikhilkumarSingh. *Wordcloud-example*. 2018. Retrieved from <https://raw.githubusercontent.com/nikhilkumarsingh/wordcloud-example/master/cloud.png>

نموذج سحابة الكلمة المعتمد على الكلام الذي يحض على الكراهية في بيئة وسائط اجتماعية عبر الإنترنت

فالتينا إبراهيم جهيدة أبو بكر نور حازلينا هارون وعلاء فريد عبداللطيف

كلية الحاسبات ، كلية الآداب والعلوم ، جامعة أوتارا ماليزيا ، سينتوك ، كيدا

الخلاصة:

تُعرف وسائل التواصل الاجتماعي باسم منصة الكاشفات التي تُستخدم لقياس أنشطة المستخدمين في العالم الحقيقي. ومع ذلك ، فإن التغذية الضخمة وغير المصفاة للرسائل المنشورة على وسائل التواصل الاجتماعي تثير تحذيرات اجتماعية ، لا سيما عندما تحتوي هذه الرسائل على خطاب كراهية تجاه فرد أو مجتمع معين. التأثير السلبي لهذه الرسائل على الأفراد أو المجتمع ككل يشكل مصدر قلق كبير للحكومات والمنظمات غير الحكومية. توفر سحابات الكلمات وسيلة بسيطة وفعالة لنقل الكلمات الأكثر شيوعاً من المستندات النصية بصرياً. يهدف هذا البحث إلى تطوير نموذج سحابة الكلمات بناءً على الكلمات البغيضة في بيئة الوسائط الاجتماعية عبر الإنترنت مثل أخبار كوكل. وقد تم اتخاذ عدة خطوات بما في ذلك الحصول على البيانات والمعالجة المسبقة واستخراج الميزات وتطوير النموذج والتصور وعرض نتيجة نموذج سحابة الكلمات. تُعرض النتائج صورة في سلسلة نصية تصف أهم الكلمات. ويمكن اعتبار هذا النموذج طريقة بسيطة لتبادل المعلومات عالية المستوى دون زيادة تحميل تفاصيل المستخدم.

الكلمات المفتاحية: جريمة الكراهية، برمجة بايثون، الشبكة الاجتماعية، سحابة الكلمات.