JIKO (Jurnal Informatika dan Komputer) Accredited KEMENDIKBUD RISTEK, No.105/E/KPT/2022
Vol. 6, No. 1, April 2023, hlm. 43-51 p-ISSN: 2614-8897
DOI: 10.33387/jiko.v6i1.5836 e-ISSN: 2656-1948

SUPPORT VECTOR MACHINE (SVM) ALGORITHM FOR STUDENT SENTIMENT ANALYSIS OF ONLINE LECTURES

Abdul Muis[1], Abdul Mubarak[2], Arifandy M Mamonto[3], Satria D Surya[4]

[1]Program Studi Teknik Informatika Universitas Mega Rezky Makassar
[2,3]Program Studi Informatika Universitas Khairun Ternate
[4]Pascasarjana Program Studi Teknologi Informasi Universitas Amikom Yogyakarta

Email: [1]abdulmuis.160674@gmail.com, [2]amuba029@unkhair.ac.id, [3]arifandymariomamonto@gmail.com
[4]satriapauwah@gmail.com

(Received: 17 February 2023, Revised: 15 March 2023, Accepted: 10 April 2023)

Abstract

Covid-19 was first discovered in Wuhan City, Hubei Province, China, at the end of December 2019. According to the WHO (World Health Organization) as of October 13 2020, the number of positive confirmed cases of Covid-19 reached 38,103,332 cases, while in Indonesia the number of cases exposed to Covid-19 reached 268.85 cases and is likely to increase every day (Covid-19 Handling Task Force, 2020). The formulation of the problem that will be raised from this research is to measure the level of accuracy obtained from the results of classifying sentiments of distance learning during the Covid-19 pandemic using the Support Vector Machine (SVM) method and measuring the impact of implementing online lectures during the Covid-19 pandemic. The data used in this research is in the form of public responses regarding distance learning policies implemented during the Covid-19 pandemic, taken from January to March 2022. The data obtained will then be divided into training data as much as 80% of the the total data and test data is, 20% of thereall data. Based on testing the previous Support Vector Machine classification model, the accuracy value for the entire system can be calculated at 70.8%. Based on the results of testing the previous Support Vector Machine classification model, the accuracy value for the entire systthe whole of thecalculated at 70.8%.

Key*words*: *COVID, WHO, Support Vector Machine (SVM), Machine learning*

**Corresponding Author: Abdul Muis*

1. INTRODUCTION

Coronavirus Disease 2019 (Covid-19) is a virus that attacks the human respiratory system and causes death, and spreads very quickly to various countries in the world). Covid-19 was first discovered in Wuhan City, Hubei Province, China at the end of December 2019. The Coronavirus then developed in sixty-five countries in February 2020 [1]. According to WHO (World Health Organization) as of October 13 2020, the number of positive confirmed cases of Covid-19 has reached 38,103,332 cases, while in Indonesia, the number of patients exposed to Covid-19 has reached 268.85 cases and is likely to increase every day (Covid-19 Handling Task Force). 19, 2020). The increase in the number of those exposed to Covid concerns for all parties. The government has issued various policies, such as independent isolation, social distancing, and physical distancing to large-scale social restrictions (PSBB) to break the chain of transmission of Covid-19.

The impact of the Covid-19 pandemic is increasingly evident in various fields, namely mental health [2], economic global [3], social tourism [4-5], and education [6-10]. Education is one of the fields most affected by the Covid-19 pandemic. Education during the pandemic must continue. To reduce the spread of Covid-19 and ensure learning activities can run as usual, the government, in this case, the Ministry of Education and Culture, is implementing a distance education system in Indonesia. Universities that previously adopted a face-to-face approach in presentations, essay guidance and other academic activities now need to change it to a distance learning approach. However, the distance learning system does not rule out differences of opinion in response to any changes that occur. One of the problems that arise from the distance learning system is that not all students have devices that support the continuity of the distance learning process. Seeing student responses to distance learning can be seen from various means and media, one of which is through social media. Social media is a source of information and media for sharing opinions and everyday life.

For this reason, the research will carry out a sentiment analysis of distance learning. Sentiment analysis of the distance learning experience was carried out by [10] with machine learning and deep learning approaches. [8] also conducted a sentiment analysis of distance learning in 10 countries during a pandemic. Indonesia is not included in the 10 countries in the study [11]. [12] applied the LSTM algorithm for sentiment analysis to distance learning.

his research will retrieve data in the form of comments on Twitter related to distance learning that occurred in Indonesia during the Covid-19 pandemic. To analyze sentiment, the support vector machine method is used. This support vector machine method has been applied to sentiment analysis in other cases [13-15]. then the method evaluation process will be carried out by applying the confusion matrix. the goal is to be able to see the performance of the SVM algorithm in conducting sentiment analysis of distance learning in Indonesia when Covid-19 occurred.

2. RESEARCH METHOD

The data used in this research is in the form of public responses regarding distance learning policies implemented during the Covid-19 pandemic, which were taken from January to March 2022. The data obtained will then be divided into training data as much as 80% of the total data, and data tests as much as 20% of the overall data.

2.1 System Design

The basic pattern of the system to be built in this study is shown in Figure 4.

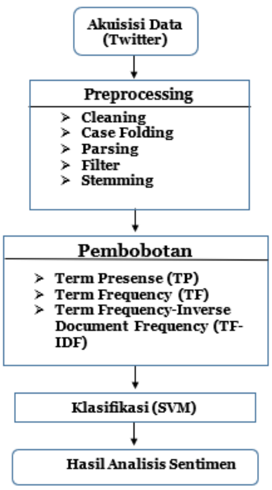

Figure 4. System Design

At this stage, data is acquired or collected from the Twitter social networking site which is connected directly to the API (Application Programming Interface) and adds a language detection process to obtain data or documents in Indonesian. In the Preprocessing stage there are several parts of stages, namely Cleansing, Case folding, parsing/tokenizing, filtering then the stemming stage to get the basic words to be classified. After that we will enter the next stage, namely weighting using the term present (TF), term frequency (TF) and term frequency-inverse document frequency (TF-IDF) methods, then the data will be classified using the Support Vector Machine (SVM) algorithm.

2.2 Preprocessing Flowchart

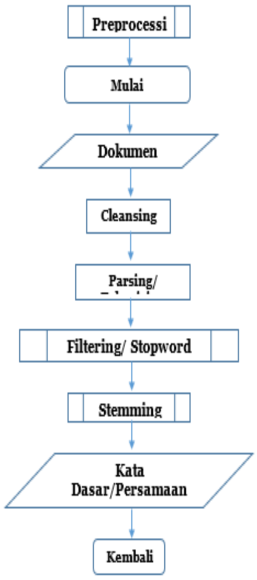

Figure 4. Flowchar tPreprocessing

At this stage, the data or documents will be extracted and entered the system, then a document cleaning process is carried out, which aims to remove characters or words that are not needed to reduce nois. After that a case folding process is carried out, namely uniform letter shapes. from A to Z, capital letters are converted to lowercase letters and then a tokenizing parsing process is carried out, namely the process where the document is broken up or divided into terms based on stopword spaces, after that a filtering/stopword removal process is carried out to filter out words or documents, after which a ssteaming process will be carried out wto obtain basic words or synonyms. This will be done repeatedly to get the base word according to KKBI

2.3 Flow diagram Filter Stopword Removal

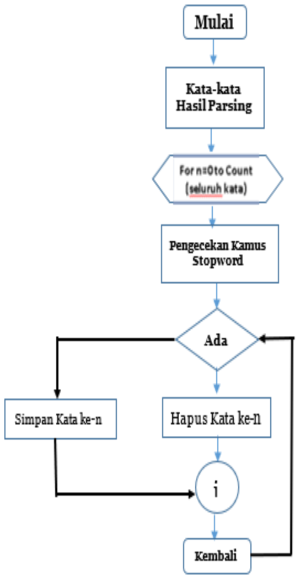

Figure 5.Flow diagramFilter Stopword Removal

At this stage the words that have been parsed/tokenized will be counted for the total number of words that exist or appear in the document then filtered or checked on the dictionary to remove irrelevant words in the document according to stopword removal, if not found these words will be deleted, but if these words are found in you then they will be stored in the database. The process of removing meaningless words. The filtering process is called Stopword Removal. At this stage using nltk. NLTK (Natural Language ToolKit) is a library provided by Python for building text analysis programs.

2.4 Process of Classification (SVM)

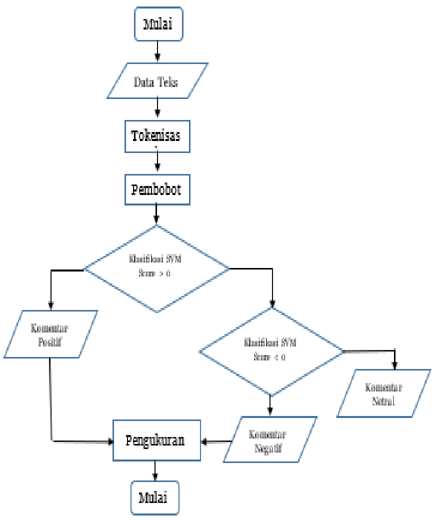

Figure 6. Classification Process (SVM)

Data or documents in the form of text or words that have gone through the Case folding, cleaning and filtering process will be tokenized and then a weighting calculation will be carried out, after weighting is classified if the word or term has a value of < 0 then the word will be entered in negative comment, but if = 0 then the word will be included in a neutral or undefined comment, and if the word is > 0 then it will be categorized in a positive comment.

In the classification system, it only looks at the point and space of the document for spatial modeling needs and the vector used then gives each word in the document to be processed, as well as the weight of the word based on how important the word is in the document. SVM in this classification process seeks to find the best line to divide the two classes and then classify the test documents based on which side the line appears.

Classification using Machine learning with the SVM algorithm will begin by changing the text into two vectors then the vector has two dimensions, namely (word id) and weight. SVM in the text classification process is only at a point in the document space so that the space model will give each word in an id document (a dimension and a weight based on how important it is in the document), the SVM method in its work tries to find the best line that divides the two classes after that do the classification of the documents tested is based on which side the line appears. SVM in the classification determines the best line that separates the two classes that have the largest margin between them.

3. RESULT AND DISCUSSION

3.1 Import library of python

The initial stage was to collect tweet data in Indonesian by searching for the keywords "kuliah online" and "kuliah daring" using the Twitter API. First do the Import Libraries provided by Python. The library used in collecting the data used is the tweepy library which can access the Twitter API directly in the console or script. The sys library is a library that is used to provide access to several variables that are used or managed by translators, the matplotlib.pyplot library is used to create visualization functions in graphical form.

```
# Import Libraries

from textblob import TextBlob
import sys
import tweepy
import matplotlib.pyplot as plt
import pandas as pd
import numpy as np
import os
import nltk
import pycountry
import re
import string
```

Figure 7. Import Library Python

3.2 Sign up on Twitter and crawling data

To be able to retrieve data from Twitter, first register at the Twitter Developer. After completing the registration process, you will be given a consumer key, consumer secret, access token and access token secret which are used to access data on Twitter.

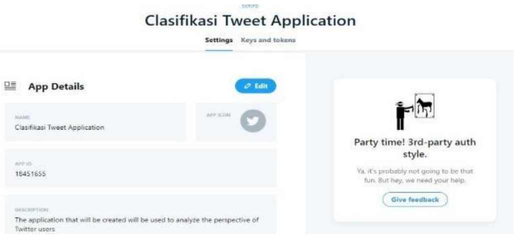

Figure 8. Sign up for Twitter Developers

After obtaining the Token from the Twitter Developer, the token authentication is then carried out with the Python code as shown in Figure 9.

```
# Authentication
consumerKey = "ptYioRiiTU1M3tB5SwEVZkcel"
consumerSecret = "y0MQO4NIueQv5nE2RFpv8Wx92dEW6JWE1UOl0VekTT7nvgmRCx"
accessToken = "815848759787749376-hfdh3LULlqR5plPwiFEg5WQGFc4sogS"
accessTokenSecret = "GrpprST61FGf1OGGl0ZBTgrFGrpWPqHNHWMP5pFX8r4aP"

auth = tweepy.OAuthHandler(consumerKey, consumerSecret)
auth.set_access_token(accessToken, accessTokenSecret)
api = tweepy.API(auth,wait_on_rate_limit=True)
```

Figure 9. Python Code of Token Authentication

After connecting to Twitter, the program will request data about the keywords and hashtags you are looking for, then enter them into the keyword variable and will ask for the amount of tweet data to be analyzed. Then it is entered into the noOfTweet variable which is an integer. The tweet variable will perform data retrieval operations that have been stored in the SearchTerm and noOfSearchTerm variables. The keywords used in the data search are "kuliah online" and "kuliah daring". The use of the Query is because are "kuliah online" and "kuliah daring" are terms used to describe learning that continues during the Covid-19 pandemic with a lecture system using technology and not face to face. Data collection was taken per day since the start of this research, namely for the period January 2022 – March 2022. This is because the Twitter API used does not support data retrieval for a long time because every day Twitter data is always update and updated automatically.

```
keyword = input("Masukkan Hashtag atau Query yang dicari: ")
noOfTweet = int(input ("Masukkan jumlah Tweet : "))


tweets = tweepy.Cursor(api.search, q=keyword, since='2021-01-01', until='2021-03-31').items(noOfTweet)
```

Figure 10. Python code of Crawling data

Data collected as many as 2000 tweets were collected and stored in a spreedsheet file with .csv format. There is no limit to the amount of data taken, but with 2000 data it is expected to represent the results of public opinion in general. The stored data consists of the date the tweet was created, Twitter users who posted tweets about "kuliah online" and "kuliah daring". The data collected in spreedsheet format can be seen in Figure 11.

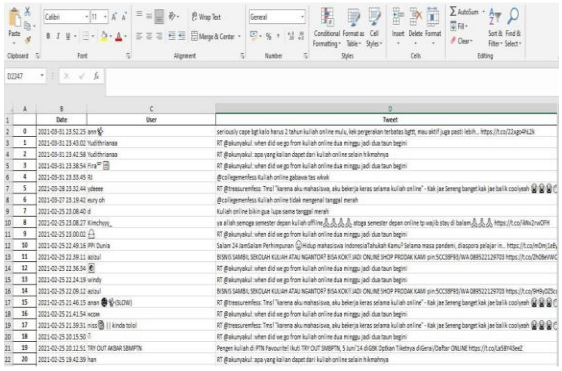
Figure 11. Results of data collection with Twitter API

The data preprocessing stage needs to be carried out because some of the tweet sentences obtained do not fully use standard words and use good Indonesian. Preprocessing is done using the help of a library in the Python programming language. Data pre-processing is carried out in the Case folding, Tokenizing, Filtering, Stemming stages so that the data is clean and ready to proceed to the next process.

3.3 Case folding

Case folding is the process of changing tweet data into lowercase. The following is an example of research data carried out by the case folding process

```
# ------ Case Folding --------
# gunakan fungsi Series.str.lower() pada Pandas
tw_list['text'] = tw_list['text'].str.lower()


print('Case Folding Result : \n')
print(tw_list['text'].head(5))
print('\n\n\n')
```

Figure 12. Python code for the Case folding stage

Table 1. Results Preprocessing of Case folding

Tweet	Case folding
Sangat mewakili dan yang mau kuliah online aja gamau offline mending keluar kampus aja sekalian dari pada ribet https://t.co/TVrd8vTkHD	sangat mewakili dan yang mau kuliah online aja gamau offline mending kelu ar kampus aja sekalian daripada ribet https://t.co/tvrd8vtkhd
@collegemenfess Kuliah gapapa offline kalo udh efektif bgt (but mungkin blm lah ya) tp gue minta bgt sidang online https://t.co/ftsoHFgiHR	@collegemenfess Kuliah gapapa offlin e kalo udh efektif bgt (but mungkin bl m lah ya) tp gue minta bgt sidang onli ne https://t.co/ftsoHFgiHR

3.4 Tokenizing

Tokenizing in this study is a stage in breaking strings or input into a text that has passed the case folding stage based on each word that composes it and eliminating URLs, @mentions and hashtags. The tokenization stage is carried out using the nltk_tokenize() function, a library in the Python3 programming language called NLTK. Import the library first which can be seen in Figure 13.

```
import string
import re #regex library
```

Figure 13.Import required library

The string library is used to load one or more characters in the tweet data. First, import the "re" library to perform the Regular Expression (regex) stages or character strings used to search text by using a pattern (pattern). Using the regex library can make it easier to find a specific string from a lot of text. In addition, at this stage the process of removing numbers, whitespace and punctuation (punctuation) is also carried out.

```
def remove_tweet_special(text):
    # remove tab, new line, ans back slice
    text = text.replace('\\t'," ").replace('\\n'," ").replace('\\u'," ").replace('\\',"")
    # remove non ASCII (emoticon, chinese word, .etc)
    text = text.encode('ascii', 'replace').decode('ascii')
    # remove mention, link, hashtag
    text = ' '.join(re.sub("([@#][A-Za-z0-9]+)|(\w+:\/\/\S+)"," ", text).split())
    # remove incomplete URL
    return text.replace("http://", " ").replace("https://", " ")

tw_list['text'] = tw_list['text'].apply(remove_tweet_special)

#remove number
def remove_number(text):
    return  re.sub(r"\d+", "", text)

tw_list['text'] = tw_list['text'].apply(remove_number)

#remove punctuation
def remove_punctuation(text):
    return text.translate(str.maketrans("","",string.punctuation))

tw_list['text'] = tw_list['text'].apply(remove_punctuation)

#remove whitespace leading & trailing
def remove_whitespace_LT(text):
    return text.strip()
```

Figure 13. Python Code of Tokenzining Level

Table 2. Preprocessing Results of Tokenizing stage

Tweet	Tokenizing
Sangat mewakili dan yang mau kuliah online aja gamau offline mending keluar kampus aja sekalian daripada ribet https: //t.co/TVrd8vTkHD @collegemenfess Kuliah gapapa offline kalo udh efektif bgt (but mungkin blm lah ya) tp gue mi nta bgt sidang online https://t. co/ftsoHFgiHR	sangat, mewakili, dan, yang, mau, kuli ah, online, aja, gamau, offline, mending, keluar, kampus, aja, sekalian, daripada, ribet kuliah, gapapa, offline, kalo, udh, efekt if, bgt, but, mungkin, blm, lah, ya, tp, g ue, minta, bgt, sidang, online

At this stage, the normalization process is also carried out, namely changing incomplete words, typing errors (typos) into normal words that can be understood properly. The results of the normalization stages can be seen in Table 3.

```
normalizad_word_dict = {}

for index, row in normalizad_word.iterrows():
    if row[0] not in normalizad_word_dict:
        normalizad_word_dict[row[0]] = row[1]

def normalized_term(document):
    return [normalizad_word_dict[term] if term in normalizad_word_dict else term for term in document]

tw_list['tweet_normalized'] = tw_list['tweet_tokens_WSW'].apply(normalized_term)
```

Figure 14. Python code of Level of Normalization

Table 3. Process Results of Normalization stage

Tweet	Normalization
kuliah, gapapa, offline, kalo, udh, efektif, mungkin, blm, tp, bgt, sidang, online	Kuliah, tidak, offline, kalau, sudah, efektif, mungkin, belum, tapi, banget, sidang, online

3.5 Filtering

The process of removing meaningless words. The filtering process is called Stopword Removal. At this stage using nltk. NLTK (Natural Language ToolKit) is a library provided by Python for building text analysis programs. First, the process of installing the nltk library is carried out. This can be done at the anaconda prompt or it can also be installed directly on the Jupyter Notebook (for those using the Jupiter Notebook idea). After the library is installed, the download process for existing stopwords is carried out and uses Indonesian stopwords. As a result, tweets that are obtained and use words that are in the Indonesian stopword list will be cleaned up.

```
import nltk
nltk.download('stopwords')

[nltk_data] Downloading package stopwords to
[nltk_data]     C:\Users\62813\AppData\Roaming\nltk_data...
[nltk_data]   Package stopwords is already up-to-date!
```

Figure 15. Python code of Import nltk

Figure 15 shows Indonesian stopword sentences provided by the nltk library.

```
from nltk.corpus import stopwords
print(set(stopwords.words('Indonesian')))

{'mungkinkah', 'antaranya', 'dikira', 'jelas', 'selamanya', 'seluruhnya', 'ke
terlaluan', 'memberikan', 'melainkan', 'menunjuknya', 'sebetulnya', 'yakni',
'kiranya', 'keadaan', 'waktu', 'kami', 'bung', 'apaan', 'cukuplah', 'adalah',
'sempat', 'dipastikan', 'toh', 'berakhirlah', 'diberikannya', 'mungkin', 'sep
antasnyalah', 'adapun', 'mau', 'berapalah', 'menyangkut', 'mempersiapkan', 's
eharusnya', 'sambil', 'tiba', 'makanya', 'ungkap', 'kira', 'berkeinginan', 't
eringat-ingat', 'sedikit', 'bersama-sama', 'mengetahui', 'bahkan', 'dini', 'd
iperbuatnya', 'satu', 'seperti', 'pasti', 'bila', 'katanya', 'akan', 'setidak
-tidaknya', 'para', 'sampai', 'diucapkan', 'bakalan', 'berujar', 'sedikitny
a', 'dijelaskan', 'lah', 'sekadarnya', 'memihak', 'sebisanya', 'menyeluruh',
'siapa', 'segala', 'setinggi', 'begitu', 'sebutnya', 'mendatang', 'setidakny
a', 'semampu', 'ditunjuknya', 'ditanya', 'sayalah', 'sudah', 'jelaslah', 'sem
ula', 'jangankan', 'maka', 'semampunya', 'menandaskan', 'masih', 'kapanpun',
'berbagai', 'itulah', 'sebelumnya', 'sepanjang', 'pukul', 'lagi', 'diperkirak
an', 'itu', 'mengungkapkan', 'mengerjakan', 'disebut', 'belakangan', 'jelaska
n', 'dijawab', 'bagaimana', 'diibaratkannya', 'mempergunakan', 'pihak', 'rupa
nya', 'maupun', 'sebagian', 'kenapa', 'pantas', 'diperlihatkan', 'terhadapny
a', 'diberikan', 'kitalah', 'dikatakan', 'pertama-tama', 'sejenak', 'bukan',
'begitupun', 'demikian', 'semuanya', 'saya', 'bersiap', 'sajalah', 'sinilah',
'diingatkan', 'kecil', 'tanpa', 'berkata', 'bertanya-tanya', 'berlangsung',
```

Figure 16. List of Stopsword Indonesia

In addition to the list of Indonesian stopwords provided by the nltk library, a list of words that are not needed in sentiment analysis is added by adding words directly to list_stopword. extends to be removed by the system. The results of the Filtering stage can be seen in Table 4.

```
from nltk.corpus import stopwords

# ----------------------- get stopword from NLTK stopword ------------------------------
# get stopword indonesia
list_stopwords = stopwords.words('indonesian')


# ---------------------------- manualy add stopword  -----------------------------------
# append additional stopword
list_stopwords.extend(["yg","gpp","make", "dg", "gua","gw", "rt", "dgn", "ny", "d", 'klo',
                       'kalo', 'amp', 'biar', 'bikin', 'bilang',
                       'gak', 'ga', 'krn', 'nya', 'nih', 'sih',
                       'si', 'tau', 'tdk', 'tuh', 'utk', 'ya',
                       'jd', 'jgn', 'sdh', 'aja', 'n', 't',
                       'nyg', 'hehe', 'pen', 'u', 'nan', 'loh', 'rt',
                       '&amp', 'yah'])

# ----------------------- add stopword from txt file ------------------------------------
# read txt stopword using pandas
txt_stopword = pd.read_csv("coba\stopwords.txt", names= ["stopwords"], header = None)

# convert stopword string to list & append additional stopword
list_stopwords.extend(txt_stopword["stopwords"][0].split(' '))

# ---------------------------------------------------------------------------------------

# convert list to dictionary
list_stopwords = set(list_stopwords)


#remove stopword pada list token
def stopwords_removal(words):
    return [word for word in words if word not in list_stopwords]

tw_list['tweet_tokens_WSW'] = tw_list['tweet_tokens'].apply(stopwords_removal)
```

Figure 17. Python code of Filtering level

Table 4. Results of Preprocessing Filtering.

tweet	filtering

Kuliah, tidak, offline, kalau, sudah,efektif, mungkin, belum, tapi, banget, sidang, online	Kuliah, offline, kalau, efektif, mung kin, belum, sidang, online

3.6 Stemming

Stemming is finding the root (base) word of each filtered word by removing the affixes in front of and after the affixes. The stemming stage is carried out using the help of a library in the Python3 programming language called Sastrawi. After the library has been installed, the next step is to import the StemmerFactory class from the Sastrawi library.

```
from Sastrawi.Stemmer.StemmerFactory import StemmerFactory
```

Figure 18. Literary Class Import

The stemming process results will delete words with affixes in prefixes, suffixes or word inserts to form basic words.

Table 5. Results of Stemming stage

Adverbs	Basic word
Represent	representative
depends	hanging
the morning	morning
course	studying
hope	front
my back	back
depressed	press
the meaning	meaning
the seller	sell
use	campus
do	wear
speak	school
	work
	talk

3.7 Data Labeling

Labeling of crawled data and having gone through the preprocessing stage is done using the Python textbob library by looking at the polarity, and subjectivity of the tweet text that has been collected. Textblob is one of the libraries provided by Python for processing in the field of Natural Language Processing which can provide word tags, word extraction, word translation and sentiment analysis. The results of labeling with textblob can be seen in Figure 19.

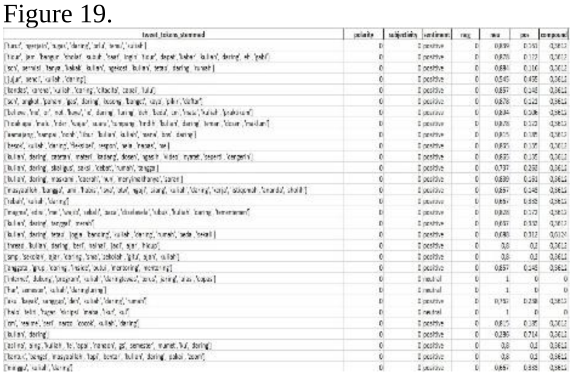

Figure 19. Labeling results of tweet withTextblob

Then the data will be divided by 80% training data and 20% test data. Sentiments on the training data of 1600 are divided manually according to class. The other 400 data will be used as test data. The results of data classification can be seen in Table 6.

Table 6. Result of Data Labeling with Textblob

Positive	Neutral	Negative	Amount
401	1368	231	2000
20,05%	68,40%	11,55%	100%

The final results obtained from labeling using the textbob library of 2000 tweet data are 401 tweets that fall into the positive class, 1368 tweets in the neutral class and 231 tweets in the negative class. The percentage of data labeling with textblob can be seen in Figure 20.

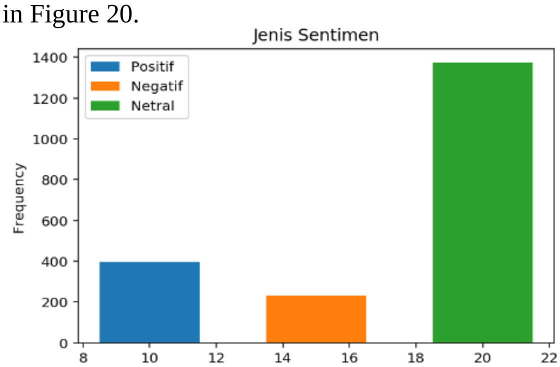

Figure 20. Percentage results for labeling data tweet with textblob

As a comparison to the results of data labeled using the textblob library, in this study manual labeling was also carried out by the author. The manual labeling process carried out by the author can be seen in Figure 21.

Data	Sentimen
allah moga semester kuliah offline atau tidak semester online tapi wajib stay balam	netral
salam jam salam himpun hidup mahasiswa indonesia tahu pandemi diaspora in	positif
bisnis sekolah kuliah kantor online shop prodak	netral
bisnis sekolah kuliah kantor online shop prodak	netral
kuliah ptn favourite ikut try out smbptn juni digbk dapatkan tiket digeraidaftar online	netral
kuliah online mana tapi liat rumah ditanyatanya kelas nyimak mam	negatif
tahun kuliah online bapak iritjuta dimintain laptop angel pol	negatif
emoji favorit semenjak kuliah online	netral
sosoaaan online lg mall tmpt wisata cafe rame sudah muak banget online sampe	netral
tanggal merah tidak ada harga pas kuliah online wes pokonyaa trabasss mlaku kuliahe	negatif
udah kuat allah kuliah online muak sekali	negatif
kuliah online kuliah online diri kuliah	negatif
tangis tolong offline tidak enak sekali kuliah online stress sekali rumah tidak kondusif	negatif
cinta kuliah online	positif
suka uji online tidak kuliah online kecuali bimbing tidak kena semprot dosen	negatif
tanggal merah kuliah online kenal tanggal merah	negatif
masyaallah mudik dipakai susah sinyal banget pakai semenjak pakai	negatif
can relate udahlah stressnya dobel dobel kuliah online tidak rumah	negatif
energi pancar mba mcd seperti kuliah online	negatif
males aku kuliah offline udah nyaman online	positif
kuliah online suka betidak begadang	negatif

Figure 21. Manual labeling by the author

Table 7. Manual data labeling results

Positive	Neutral	Negative	Amount
416	1262	322	2000
20,8%	63,1%	16,1 %	100%

Table 7 shows the results of manual labeling carried out by the author.

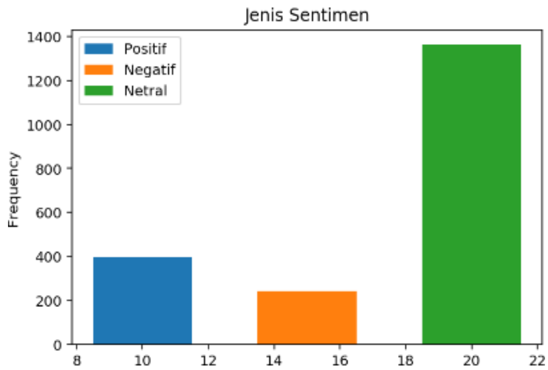

Figure 21. The results of the percentage of manual data labeling

There are differences in the results of the classification with the textblob library and the results of manual classification carried out by the author. The difference in classification results between labeling using textblob and manual labeling carried out by the author is 0.75% in the positive class, 5.3% in the neutral class and 4.55% in the negative class.

3.8 Feature Extraction

After the tweet data has gone through the pre-processing process, the next data will be formed into a classification model. Before making a model there are several steps that need to be done in order to produce a good model. In the feature extraction stage, the first step is to convert the research dataset into a vector representation. Python has a named Library which can be used in Machine learning. This library contains the CountVectorizer algorithm which can convert text features into a vector representation. The results of the vector representation are 2000 numbers which have 4092 words.

3.9 Model build

After going through the preprocessing of the data and the vectorizer, a model is then created that will be used to classify the test data.

3.9.1 Distribution of data test

After going through the pre-processing of the data and the vectorizer, a model is then created that will be used to classify the test data.

3.9.2 SVM Model Implementation

In order to implement the model, we will use the help of a library in the Python3 programming language called scikit-learn which contains SVM in this library and from here we will call SVC for the model.

3.9.3 Model Testing and Accuracy

In order to find out the performance of the Support Vector Machine (SVM) Algorithm, a test was carried out on the model that has been made along with its accuracy. this is done to measure the performance of a Support Vector Machine (SVM) classification method.

3.9.4 Classification and Confusion Matrix Plotting

Basically the confusion matrix contains information that compares the results of the classification performed by the system with the results of the classification that should be. The following is a classification model using the Python sklearn library. Metrics included has a confusion_matrix and is visualized using seaborn which is an open source visualization library built on top of the matplotlib library.

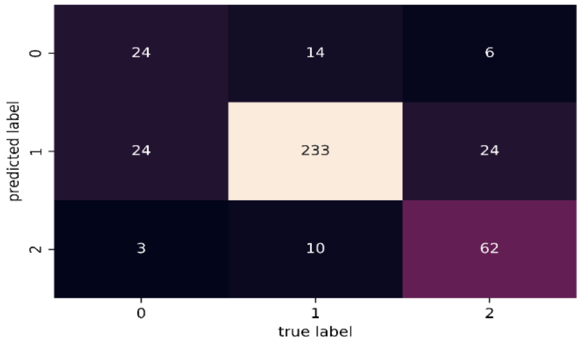

Figure 22. Confusion Matrix Visualization

3.10 Model Evaluation

After testing the model is complete, the next step is to evaluate the model. Model evaluation aims to produce a confusion matrix with a size of 3 x 3. The confusion matrix provides comparative information on the classification results performed by the classification model with the actual classification results.

		Predict Class		
		Positif	Netral	Negatif
	Positif	24	14	6
Actual Class	Netral	24	233	24
	Negatif	3	10	62

Figure 23. The Confusion Matrix Results

Based on the results of testing the previous Support Vector Machine classification model, the accuracy value for the entire system can be calculated at 70.8%. Calculation of manual accuracy of matrix calculations is as follows:

$$\text{Akurasi} = \frac{\text{True Positif} + \text{True Netral} + \text{True Negatif}}{\textit{Total Data yang di Uji}} \times 100\,\%$$

$$= \frac{319}{400}$$

$$= 70{,}8\ \%$$

Figure 24. The Accuracy value

Accuracy describes how much accurate the model that has been made can classify the data correctly. Accuracy is obtained from calculating the ratio of correct predictions to all data. By knowing the magnitude of the accuracy value on the overall performance of the system, it can be stated that the level of the system's ability to find the accuracy between the information that the user wants and the answers given by the system. The success rate of the system in finding information in this study is 70.8%.

Furthermore, to see the classification performance of each class can be known through the value of precision, recall and f1 score in each classification class. Precision describes the level of accuracy of the requested data with the results provided by the model. Precision is obtained from calculating the ratio of correct predictions compared to the overall positive predicted results. Recall describes the success of the model in retrieving the information included in the test. Recall is obtained from the calculation of the ratio of true positive predictions compared to all data that is positive. F1-Score is a single parameter measure of retrieval success that combines Recall and Precision.

The results of the precision, recall, and F1-score values have a value of 0-1. The higher the value, the better the results of the model made. A high accuracy value is obtained when a lot of data has been classified correctly according to the sentiment class. You can also find out the Precision and Recall values.

The Precision value follows the accuracy value, the higher the accuracy value, the higher the Precision value will follow, and vice versa. The Precision value is the number of positive data that is correctly classified as positive data divided by the total data that are classified as positive data. While the Recall Value is the number of positive data that is correctly classified as positive data divided by the number of actual positive data. In the previous confusion matrix, you can know the True positive and True negative values. True positive is a positive data value that is correctly classified according to the sentiment class, namely positive. True negative is a sentiment data value correctly classified according to sentiment class.

Jenis Klasifikasi	Presisi	Recall	F1-Score
Positif	0,55	0,47	0,51
Netral	0,83	0,91	0,87
Negatif	0,83	0,67	0,74

Figure 25. Precision Value, Recall and F1-Score Model Evaluation

The precision value for the positive class is 55%, for the neutral class is 83%, for the negative class is 83%. This figure means that the proportion of labels predicted correctly from the total predictions is quite high for neutral and negative classes. While the system's success rate in retrieving information for the positive class is 47%, for the neutral type is 91%, and for the negative class is 67%. This means that the performance of the system's success in retrieving positive information in documents is low compared to finding negative and neutral information. Obtained an avdision value of 0.79%, a Recall value of 80% and F1-Score value of 79%.

4. CONCLUSION

Based on the results of testing the Support Vector Machine (SVM) algorithm that has been carried out, several things have been produced, including:

In this study, the Support Vector Machine (SVM) algorithm is proven to be an accurate algorithm because it produces an accuracy value of 70.8%The results of Twitter sentiment analysis with the keywords online lectures and online lectures in this study have a Precision value of 0.79%, Recall value of 80% and F1-Score value of 79%.

Based on the results of an analysis of the patterns contained in the research data, the impact of implementing online lectures during the Covid-19 pandemic is a) Students have difficulties in the internet network b) The emergence of psychological problems in students due to the increasing number of assignments and the lack of interaction with fellow students causes emotional instability and triggers a sense of stress, c) The continuous use of technological devices makes technology more quickly damaged d) The lack of lecturer knowledge regarding the technology used during online learning causes the lecture process to be slightly hampered e) Delivery material that is difficult for students to understand if they don't meet face to face, but the assignment of more and more assignments makes students unable to continue to carry out online lectures.

5. REFERENCE

[1] N. Astini, "Utilization of Information Technology in elementary school level learning during the Covid-19 pandemic,"*The Light Journal*.vol. 11(2), p. 13-25, 2020.

[2] Giuntella, O. dkk., 2021. "Lifestyle and mental health disruptions during COVID-19", *PNAS*, vol. 118, no. 9, pp. 1–9, doi: https://doi.org/10.1073/pnas.2016632118

[3] Zhilenko. V. Y. dkk., 2021. "The Impact of COVID-19 Pandemic on the Global Economy and Environment ", *JEMT*, vol. 12, no. 5, doi: https://doi.org/10.14505//jemt.v12.5(53).08

[4] Abbas. J. dkk., 2021. "Exploring the impact of COVID-19 on tourism: transformational potential and implications for a sustainable recovery of the travel and leisure industry ", *Current Research in Behavioral Sciences*, vol. 2, 100033 doi: https://doi.org/10.1016/j.crbeha.2021.100033

[5] Restikadewi. A. dkk., 2021. "The impact of COVID-19 on the tourism sector in Indonesia", *Sebelas Maret Business Review*, vol. 6, No 1. Pp. 47-56

[6] Rulandari. N., 2020. "The Impact of the Covid-19 Pandemic on the World of Education in Indonesia", *ilomata international journal of social sciene*, vol. 1, No. 4. doi: https://doi.org/10.52728/ijss.v1i4.174

[7] Indawati. N. N., 2022. "Dampak Pandemi Covid-19 terhadap Aktivitas Pembelajaran Daring", *JPIG (Jurnal Pendidikan dan Ilmu Geografi)*, vol. 7, No. 2. doi: https://doi.org/10.21067/jpig.v7i2.7493

[8] Septiadi. M. A., dkk. 2022. "Dampak Covid-19 Terhadap Sistem Pendidikan", *Khazanah Pendidikan Islam*, vol. 4, No. 2. Pp. 51-62. DOI: 10.15575/kp.v4i1.17891

[9] Tristaningrat. M. A. N., dkk. 2021. "DAMPAK

PANDEMI COVID-19 BAGI PENDIDIKAN INDONESIA", *Maha Widya Bhuawana*, vol. 4, No. 2. DOI: https://doi.org/10.55115/bhuwana.v4i2.1760

[10] OSMANOGLU. U. O., 2020. "Sentiment Analysis for Distance Education Course Materials: A Machine Learning Approach", *ilomata international journal of social sciene*, vol. 3, No. 1. Doi: 10.31681/jetol.663733

[11] Arambepola. N., 2020. "Analysing the Tweets about Distance Learning during COVID-19 Pandemic using Sentiment Analysis", *Proceedings International Conference on Advances in Computing and Technology (ICACT–2020)*

[12] Waheeb. S. A., 2022. "Topic Modeling and Sentiment Analysis of Online Education in the COVID-19 Era Using Social Networks Based Datasets", *Electronics.* Vol 11. 715. pp, 1-20 doi: https://doi.org/10.3390/electronics11050715

[13] Oktafani, M., dan Prasetyaningrum, P. T., 2022. "Implementasi Support Vector Machine Untuk Analisis Sentimen Komentar Aplikasi Tanda Tangan Digital", *Jurnal Sistem Informasi dan Bisnis Cerdas (SIBC).* Vol 15. No 1. pp, 10-19 doi: https://doi.org/10.33005/sibc.v15i1.2697

[14] Darwis. D., 2020. "ENERAPAN ALGORITMA SVM UNTUK ANALISIS SENTIMEN PADA DATA TWITTER KOMISI PEMBERANTASAN KORUPSI REPUBLIK INDONESIA", *Jurnal Ilmiah Edutic/.* Vol 7. No. 1. pp, 1-11.

[15] Waheeb. S. A., 2022. "Analisis Sentimen Terhadap Ulasan Aplikasi Pejabat Pengelola Informasi dan Dokumentasi Kementerian Dalam Negeri Republik Indonesia di Google Playstore Menggunakan Metode Support Vector Machine", *Jurnal Teknologi dan Komunikasi Pemerintahan*. Vol 4. No 1. pp, 1-13