



Comparison of the Naïve Bayes Method and Support Vector Machine in Sentiment Analysis of Genshin Impact Game Reviews

Mr. Aldiansyah Pratama¹, Fuad Nur Hasan²

^{1,2}Informatics Facilities Development University, Indonesia

ARTICLE INFO

Article history:

Received May 27, 2024

Revised Aug 15, 2024

Accepted Aug 30, 2024

Keywords:

Naïve Bayes;
Sentiment Analysis;
Support Vector Machine.

ABSTRACT

Genshin Impact was a successful and quite popular game during the 4 years of its release, but behind this there are several positive or negative opinions about this game, both internal and external. Sentiment Analysis is a technique that can identify an opinion in a text that is managed, be it a comment or review. The aim of the research is to compare two algorithms, namely Support Vector Machine and Naïve Bayes, in classifying Genshin Impact game reviews on Google Playstore. This method has several stages, namely crawling data, text preprocessing, using a confusion matrix and k-fold cross validation, all of these stages are carried out using libraries in Python with 1198 review data divided between test data and training data by 90:10 which produces a support vector machine of 73% accuracy, 75% precision, 64% recall and f1-score of 64% while naïve bayes is 72% accuracy, 68% precision, 68% recall and f1-score of 68%. With this comparison it is concluded that support vector machine has a higher evaluation value than naïve bayes, while it is known that the majority of review data has a negative value regarding Genshin Impact game reviews.

This is an open access article under the [CC BY-NC](https://creativecommons.org/licenses/by-nc/4.0/) license.



Corresponding Author:

Mr. Aldiansyah Pratama,
Informatics Facilities Development University,
Indonesia.
Email: m4ldiansyah32@gmail.com

1. INTRODUCTION

The rapid development of information technology in the digital era has resulted in many new things in aspects of daily life, one of which is online games which have become a favorite in the post-Covid-19 era due to disruptions in normal activities, forcing many people into Stay-at Home and quarantine policies. making many people shift their activities to consuming digital media entertainment, even though now in the post-Covid-19 era this habit has become a new daily habit for many people. "Games are a popular form of digital media entertainment today, people usually check game reviews and ratings before playing on their devices." (King et al., 2020) "Genshin Impact is one of the current popular games which was released in 2020, but its popularity with some people liking this game means that others are dissatisfied with the game." (Primandani Arsi, Pungkas Subarkah, & Bagus Adhi Kusuma, 2023) In this case, this research addresses whether the Genshin Impact game has a negative or positive view from a consumer perspective based on evaluations of Genshin Impact which has a bad image on social media using sentiment analysis with two algorithm methods namely Naïve Bayes and Support Vector Machine (SVM).

"Sentiment analysis is the process of identifying and grouping text reviews into sentiments which can be positive, neutral or negative." (Kusnadi et al., 2021) By comparing two algorithm methods, namely Naïve Bayes and Support Vector Machine (SVM), it is hoped that we can find out

how these two methods can be implemented in knowing the current state of the game based on existing reviews by comparing the accuracy of ratings and reviews. Therefore, this research can be a key point for researchers and technicians in the area of IT, mainly in the development of analytical models that are intended for sentiment analysis that can be used in many fields like online marketing, customer care, and social media analysis. The main contributions of this research to the field of science are: a) Achieving a major breakthrough in the comparison between the two main approaches within the sentiment analysis, the Naive Bayes and the Support Vector Machine (SVM) and thus making it possible to carry out the research in the area of Artificial Intelligence and Machine learning alike. b) Giving a basic lesson on data crawling and text pre-processing from game reviews and providing a solution to manage a huge number of user reviews. c) Addressing the issue of mobile app developers, technology companies as well as spreading the knowledge among technology students and by establishing a quality besides swiftness in sentiment analysis not only through other domains of e-commerce, digital entertainment and online service review but also in the rest of the areas of them.

2. RESEARCH METHOD

Metode penelitian yang digunakan dalam penelitian ini adalah metode penelitian kuantitatif yang menggunakan data primer yang berasal dari ulasan dan rating pengguna, karena banyaknya data hasil ulasan pada playstore pada dengan menggunakan data hasil crawling data. Menurut Emzir mendefinisikan pendekatan kuantitatif adalah satu pendekatan yang secara primer menggunakan paradigma postpositivist dalam mengembangkan ilmu pengetahuan (seperti pemikiran tentang sebab akibat, reduksi kepada variabel, hipotesis dan pertanyaan spesifik menggunakan pengukuran dan observasi serta pengujian teori), menggunakan strategi penelitian seperti eksperimen dan survei yang memerlukan data statistik. Sehingga dalam penelitian kuantitatif, sesuai dengan namanya banyak dituntut menggunakan angka, mulai dari pengumpulan data, penafsiran terhadap data tersebut, serta penampilan dari hasilnya. (Ph.D. Ummul Aiman et al., 2022).

3. RESULTS AND DISCUSSIONS

This chapter will explain the process and results of sentiment analysis taken from user reviews of the Genshin Impact game from the Google Playstore application as many as 1,198 data. The data is processed into training data as many as 1,078 training data and test data as many as 120 data. The algorithms that will be used in sentiment analysis are Support Vector Machine and Naive Bayes.

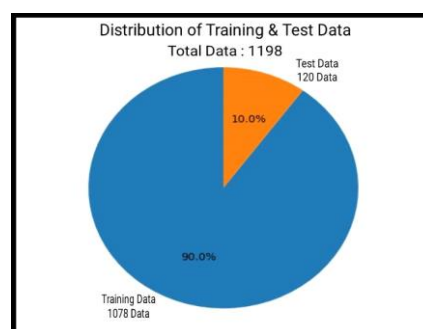


Figure 1. Distribution of Training

In this discussion, we will explain in detail the data collection process, data processing, algorithm methods used in conducting the analysis and the results of sentiment analysis as follows:

1) Sample

In this stage, to obtain samples, namely data sources in this study originating from the Google Play Store, data crawling is carried out using a library found in Python, namely the Google Play Scraper, which can access the Google Play Store API. The following are the steps in conducting data crawling:

- Do the coding by importing several libraries to help in crawling data as below.

```
!pip install -q google_play_scraper #data mining
!pip install -q transformers #sentimen
!pip install -q plotly-express #data visual

[ ] import json
import pandas as pd
from tqdm import tqdm
from google_play_scraper import Sort, reviews, app
import numpy as np

from pygments import highlight
from pygments.lexers import JsonLexer
from pygments.formatters import TerminalFormatter

[ ] from google.colab import drive
drive.mount('/content/gdrive', force_remount=True)

path = "gdrive/MyDrive/"

Mounted at /content/gdrive
```

Figure 2. Processing API

- Then enter the coding to save all the crawled data into Google Drive.
- Now determine the application that we will take the review on the Playstore where this research will focus on the Genshin Impact application where we will set several parameters such as only taking reviews from Indonesia in Indonesian, which are arranged based on the most relevant or latest reviews where we will take data as much as 1198 review data.

```
[ ] app_infos = []
for application in tqdm(app_packages):
    info = app(application, lang='id', country='id')
    get_info['comments']
    app_infos.append(info)

100% [██████████] 1/1 [00:00<00:00, 5.07it/s]

[ ] app_reviews = []
for ap in tqdm(app_packages):
    for score in list(range(1,6)):
        for sort_order in [Sort.MOST_RELEVANT, Sort.NEWEST]:
            rvs, _ = reviews(
                ap,
                lang='id',
                country='id',
                sort=sort_order,
                count=200 if score == 3 else 100,
                filter_score_with = score
            )
            for r in rvs:
                r['sortOrder'] = 'most_relevant' if sort_order == Sort.MOST_RELEVANT else 'newest'
                r['appId'] = ap
            app_reviews.extend(rvs)

100% [██████████] 1/1 [00:02<00:00, 2.93s/it]
```

Figure 3. Review Playstore

- To check the data obtained, you can use the code as shown in the image below.

```
df_busu = pd.DataFrame(np.array(app_reviews), columns=['review'])
df_busu = df_busu.join(pd.DataFrame(df_busu.pop('review').tolist()))
df_busu.head()
```

	reviewId	userName	userImage	content	score	thumbsUpCount	reviewCreatedVersion	at	replyContent	repliedAt	appVersion
0	0040d24-477c-4205-b209-42bebacad465	Adriano Tegap	https://play.googleusercontent.com/4ALVU...	Farming artikel 2 bukan untuk 1 karakter belum...	1	367	4.0_21307647_21321795	2024-03-26 07:01:36	None	Nat	4.0_21307647_21321795
1	01460291-3397-844a-89c5-74f23c16a8d	Calvin Septian Syahputra	https://play.googleusercontent.com/4ALVU...	Jujur saya kecewa dengan grafik developer BGSK...	1	338	4.4_20481013_20559831	2024-02-22 14:03:11	None	Nat	4.4_20481013_20559831
2	ba5ee05-3569-4168-b48f-a293e18d511	Mashudi Munandar 9913	https://play.googleusercontent.com/4ALVU...	Bunuh Mur bual baik ke diang terakhir di m...	1	186	4.0_21307647_21321795	2024-03-16 14:18:00	None	Nat	4.0_21307647_21321795
3	19179279-f763-4300-9662-ef70c6e86098	Rahayu	https://play.googleusercontent.com/4ALVU...	Tidak ada perubahan pada stur game selama 21...	1	628	4.3_19700728_19706476	2024-01-24 15:01:31	None	Nat	4.3_19700728_19706476
4	9e544343-6207-XI-B-32-Mikael-Christyo Yulistra Pandaniama	Christyo Yulistra Pandaniama	https://play.googleusercontent.com/4ALVU...	Tolong sistem 3030 abehukan, saya ex kasan...	1	5	4.0_21307647_21321795	2024-04-12 13:11:32	None	Nat	4.0_21307647_21321795

Figure 4. Dataframe

- From the image above, it is known that the reviews obtained have met the needs to be used as a data source in this study, after which a check was carried out on how much data was obtained as shown in the image below.

```
[ ] len(df_busu.index)
↔ 1198
```

Figure 5. Dataframe len

- With the picture above, it is known that the data that has been taken is 1198 reviews, followed by taking the required column categories where we will focus on the content and score of the column and here we will label positive and negative reviews based on the value of the score where the score is below 4, namely 1, 2 and 3 will be labeled as negative and positive will be given to values 4 and 5 as shown below.

```
[ ] my_df=my_df[['content', 'score']]

[ ] def pelabelan(score):
    if score < 4:
        return 'Negatif'
    elif score == 4:
        return 'Positif'
    elif score == 5:
        return 'Positif'
my_df['Label'] = my_df['score'].apply(pelabelan)
my_df.head(50)
```

Figure 6. Data Label

- Once completed, it will produce some data that you can see as shown in the image below.

```
***
      content  score  Label
0  The most exciting game I've ever played, hopef...  5  Positif
1                      Good  5  Positif
2                      Good  4  Positif
3  Right flat graph emg the best  5  Positif
4  A Masterpiece  5  Positif
5  Miserly Game 🙄👎👎👎  1  Negatif
6  okay bangeeett, just too heavy ... and also th...  4  Positif
7                      Love  5  Positif
8  There is a little bag  5  Positif
9                      Cool  5  Positif
10 go die lol u didn't deserve to live (real) , o...  4  Positif
11 waw  5  Positif
12 Nice game  5  Positif
13 Story is cool, gameplay is also interesting, o...  4  Positif
14 A very eye-pleasing game, with a thrilling sto...  5  Positif
15 thank you genshin for being the best friend!!  5  Positif
16 Great game  5  Positif
17 The social assistance is very small 🙄  4  Positif
18 The game is also fun.  5  Positif
19 Hope be better in reward but this game is good...  5  Positif
20 Fun  5  Positif
```

Figure 7. Data Preview

- The final step is to save the data so that it can be used in the next stage.

```
***
      content  score  Label
0  The most exciting game I've ever played, hopef...  5  Positif
1                      Good  5  Positif
2                      Good  4  Positif
3  Right flat graph emg the best  5  Positif
4  A Masterpiece  5  Positif
5  Miserly Game 🙄👎👎👎  1  Negatif
6  okay bangeeett, just too heavy ... and also th...  4  Positif
7                      Love  5  Positif
8  There is a little bag  5  Positif
9                      Cool  5  Positif
10 go die lol u didn't deserve to live (real) , o...  4  Positif
11 waw  5  Positif
12 Nice game  5  Positif
13 Story is cool, gameplay is also interesting, o...  4  Positif
14 A very eye-pleasing game, with a thrilling sto...  5  Positif
15 thank you genshin for being the best friend!!  5  Positif
16 Great game  5  Positif
17 The social assistance is very small 🙄  4  Positif
18 The game is also fun.  5  Positif
19 Hope be better in reward but this game is good...  5  Positif
20 Fun  5  Positif
```

Figure 8. Data preview

2) Explore

The following table is an example of the results of cleaning review data.
Post Cleaning

Table 1. Comparison

Original Review	Cleaning Reviews
The most exciting game I've ever played, hopefully the developer will increase the social assistance because there is a shortage of Primo for the Arlecchino gacha later (pls developer be kind to us)	The most exciting game I've ever played. I hope the developer will increase the social assistance because there is a shortage of Primo for gacha arlecchino later. Pls, developers, be kind to us.

5) **Case Folding**

Folding case is a process that will change all capital letters in the review data to lower case using a code like the image below.

```
my_df['text_clean'] = my_df['content'].str.lower()
my_df['text_clean']
data_clean = clean_text(my_df, 'content', 'text_clean')
data_clean.head(10)
```

	content	score	Label	text_clean
0	Game paling seru yg pernah gw mainin, semoga a...	5	Positif	game paling seru yg pernah gw mainin semoga aj...
1	Baik	5	Positif	baik
2	Bagus	4	Positif	bagus
3	Grafik rata kanan emng the best	5	Positif	grafik rata kanan emng the best
4	Sebuah Mahakarya	5	Positif	sebuah mahakarya
5	Game Klor 🤬🤬🤬🤬	1	Negatif	game klor
6	oke bangeett, cuman terlalu berat... dan juga...	4	Positif	oke bangeett cuman terlalu berat dan juga ban...
7	Love	5	Positif	love
8	Ada sedikit bag	5	Positif	ada sedikit bag
9	Keren	5	Positif	keren

Figure 12. Case Folding

In this case, data that has been cleaned will be changed from uppercase to lowercase to minimize system reading errors due to letter size sensitivity, an example of this case is as in the table below.

Post Case Folding

Table 2. Post Case Folding

Cleaning Reviews	Folding Case Review
The most exciting game I've ever played. I hope the developer will increase the social assistance because there is a shortage of Primo for gacha arlecchino later. Pls, developers, be kind to us.	The most exciting game I've ever played. Hopefully the developer will increase the social assistance because there is a shortage of primo for gacha arlecchino later. Pls developers be kind to us.

6) **Stopword Removal**

The next stage aims to remove common words that often appear in review data but do not influence sentiment in a sentence, using one of the libraries in Python, namely the NLTK Corpus stopwords Indonesian language as in the code below.

```
import nltk.corpus
nltk.download('stopwords')
from nltk.corpus import stopwords
stop = stopwords.words('Indonesian')
data_clean['text_stopword'] = data_clean['text_clean'].apply(lambda x: ' '.join(word for word in x.split() if word not in (stop)))
data_clean.head(50)
```

	content	score	Label	text_clean	text_stopword
0	Game paling seru yg pernah gw mainin, semoga a...	5	Positif	game paling seru yg pernah gw mainin semoga aj...	game seru yg gw mainin semoga apa perbanyak ba...
1	Baik	5	Positif	baik	bagus
2	Bagus	4	Positif	bagus	bagus
3	Grafik rata kanan emng the best	5	Positif	grafik rata kanan emng the best	grafik kanan emng the best
4	Sebuah Mahakarya	5	Positif	sebuah mahakarya	mahakarya
5	Game Klor 🤬🤬🤬🤬	1	Negatif	game klor	game klor

Figure 13. Stopword Removal

Post Stopword Removal

Table 3. Post stopword Removal

Folding Case Review	Stopword Removal Review
The most exciting game I've ever played. I hope the developer will increase the social assistance because there is a shortage of Primo for gacha arlecchino later. Pls, developers, be kind to us.	fun game that I play, hopefully the developer will increase social assistance because there will be a shortage of primo gacha arlecchino later, please developers be kind

7) Tokenize

This stage will make the review data of the previous case folding results broken down into pieces of words or often called tokens, below is the code for tokenizing and the results are tables that are examples of changes when tokenizing.

```
import nltk
nltk.download('punkt')
from nltk.tokenize import sent_tokenize, word_tokenize
data_clean['text_tokens'] = data_clean['text_StopWord'].apply(lambda x: word_tokenize(x))
data_clean.head()
```

15	makasih genshin sudah jadi teman terbaik 🎮	5	Positif	makasih genshin sudah jadi teman terbaik	makasih genshin teman terbaik	[makasih, genshin, teman, terbaik]
16	Mantap game nya	5	Positif	mantap game nya	mantap game nya	[mantap, game, nya]
17	Bansos nya dikit amat 🍷	4	Positif	bansos nya dikit amat	bansos nya dikit	[bansos, nya, dikit]
18	Seru juga game nya .	5	Positif	seru juga game nya	seru game nya	[seru, game, nya]

Figure 14. Tokenize

Post Tokenizing

Table 4. Post Tokenizing

Stopword Removal Review	Tokenize Review
fun game that I play, hopefully the developer will increase social assistance because there will be a shortage of primo gacha arlecchino later, please developers be kind	['game', 'fun', 'yg', 'gw', 'play', 'hopefully', 'just', 'increase', 'social assistance', 'developer', 'nya', 'because', 'lack', 'primo', 'gacha', 'arlecchino', 'later', 'pls', 'developer', 'be kind', 'be careful']

8) Stemming

Stemming is the final process in text preprocessing which simplifies existing words into basic words of the word by removing the affixes in the word, the reason for this process is to standardize the words so that the vocabulary can be read in the same form.

This stemming process uses the Sastrawi library in Python programming which helps in the process using the code in the image below and the table as an example of stemming results.

```
#-----STEMMING-----
from Sastrawi.Stemmer.StemmerFactory import StemmerFactory

factory = StemmerFactory()
stemmer = factory.create_stemmer()

def stemmed_wrapper(term):
    return stemmer.stem(term)

term_dict = {}
hitung=0

for document in data_clean['text_tokens']:
    for term in document:
        if term not in term_dict:
            term_dict[term] = ''

print(len(term_dict))
print("-----")
for term in term_dict:
    term_dict[term] = stemmed_wrapper(term)
    hitung+=1
    print(hitung,":",term,":", term_dict[term])

print(term_dict)
print("-----")

def get_stemmed_term(document):
    return [term_dict[term] for term in document]

data_clean['text_stemindo'] = data_clean['text_tokens'].apply(lambda x: ''.join(get_stemmed_term(x)))
data_clean.head(20)
```

Figure 15. Stemming

testing using the K-Fold Cross Validation method 10 times, based on research. (Jimmy Alga, Cindi Wulandari, 2024).

4) Assess

Assess is a stage that provides evaluation results from the research model that is currently being studied using the naïve Bayes method or support vector machine, then the evaluation results use a confusion matrix with accuracy, precision, recall, and f-1 score values from the review data, in maximizing its value, k-folds cross validation is used with a value of $k = 10$.

5) Naïve Bayes Method

In the Naïve Bayes method, a data division scenario of 90%: 10% is used between training data and test data, which is carried out 10 times with the k-folds cross validation method to obtain the highest results in data testing. The following data is the result of k-folds cross validation using naïve bayes in the form of images of accuracy, precision, recall and f1-score.

```

-----GENSHIN Naive Bayes-----
fit_time score_time test_accuracy test_precision_macro \
0 0.012964 0.012967 0.712963 0.738324
1 0.006981 0.051860 0.768519 0.708882
2 0.004987 0.015958 0.759259 0.720513
3 0.008977 0.038896 0.731481 0.722222
4 0.016953 0.013963 0.712963 0.675028
5 0.003989 0.008978 0.712963 0.644533
6 0.002992 0.009973 0.759259 0.726304
7 0.003989 0.006981 0.657407 0.633117
8 0.005986 0.006980 0.672897 0.628617
9 0.003990 0.009975 0.775701 0.734330

test_recall_macro test_f1_macro
0 0.689583 0.687191
1 0.721737 0.714437
2 0.705087 0.711349
3 0.723722 0.722905
4 0.687202 0.679200
5 0.647436 0.645902
6 0.732681 0.729167
7 0.637124 0.634568
8 0.609846 0.613320
9 0.705625 0.716180

```

Figure 19. Naïve bayes method

6) Support Vector Machine (SVM) Method

Then, with the Support Vector Machine (SVM) method which uses the same scenario, namely 90%: 10% between training data and test data, then carried out 10 times with the k-folds validation method, the following data results using SVM are in the form of images of accuracy, precision, recall, and f1-score.

```

-----GENSHIN SVM-----
fit_time score_time test_accuracy test_precision_macro \
0 0.116489 0.022043 0.796296 0.828804
1 0.118557 0.018747 0.750000 0.743011
2 0.107277 0.019672 0.787037 0.745924
3 0.117831 0.018664 0.777778 0.796380
4 0.110062 0.021940 0.703704 0.675595
5 0.110120 0.018819 0.731481 0.754559
6 0.116235 0.018190 0.712963 0.721505
7 0.111895 0.019659 0.750000 0.825988
8 0.114604 0.017950 0.728972 0.774561
9 0.102841 0.018738 0.728972 0.756322

test_recall_macro test_f1_macro
0 0.692369 0.713735
1 0.634738 0.643651
2 0.658010 0.677193
3 0.679452 0.696060
4 0.633083 0.638039
5 0.627522 0.627807
6 0.616165 0.612455
7 0.659420 0.662461
8 0.620849 0.616203
9 0.664819 0.667524

```

Figure 20. SVM method

7) Naïve Bayes Method

With the results of using the confusion matrix with the naïve Bayes method at a ratio of 90%: 10% with the test data results obtained 72 negative test data and 19 positive values from 120 test data with the following calculations:

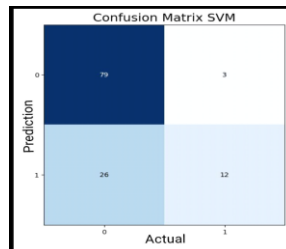


Figure 21. Confussion matrix SVM

The precision calculation for positive and negative classes is as per the calculation below: Then there is a recall consisting of positive and negative class recall with the calculation below:

The last one is the f1-score calculation as follows:

And the results of accuracy, precision, recall and f1-score are in accordance with Figure IV.24. The results of these values after the calculation obtained the highest data from k-fold cross validation are in row 10 with 77% accuracy, 73% precision, 70% recall and f1-score 71% as in the table below:

Table 6. Naïve Bayes K-Fold Cross Validation Results

No	Accuracy	Precision	Recall	F1-Score
1	71%	73%	68%	68%
2	76%	70%	72%	71%
3	75%	72%	70%	71%
4	73%	72%	72%	72%
5	71%	67%	68%	67%
6	71%	64%	64%	64%
7	75%	72%	73%	72%
8	65%	63%	63%	63%
9	67%	62%	60%	61%
10	77%	73%	70%	71%

- Support Vector Machine Method

Now by using the support vector machine method with a ratio of 90%: 10% with the test data results obtained 75 test data with negative values and 12 positive values from 120 test data with the following calculations:

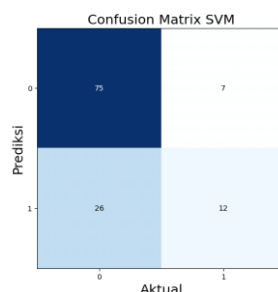


Figure 22. Confussion matrix SVM

The precision calculation for positive and negative classes is as per the calculation below: Then there is a recall consisting of positive and negative class recall with the calculation below: The last one is the f1-score calculation as follows:

And the results of accuracy, precision, recall and f1-score are in accordance with Figure IV.25. The results of these values after the calculation obtained the highest data from k-fold cross

validation are in row 2 with 80% accuracy, 79% precision, 67% recall and f1-score 69% as in the table below:

Table 7. K-Fold Cross Validation Support Vector Machine Results

No	Accuracy	Precision	Recall	F1-Score
1	62%	71%	58%	53%
2	80%	79%	67%	69%
3	75%	75%	62%	63%
4	77%	81%	73%	74%
5	76%	83%	64%	64%
6	80%	78%	69%	71%
7	78%	80%	69%	71%
8	61%	53%	52%	50%
9	71%	71%	62%	62%
10	79%	82%	67%	69%

8) Comparison

In the results of the two methods in this study, namely Naïve Bayes and Support Vector Machine, a performance comparison will be carried out on aspects of accuracy, precision, recall and F1-score after finding the average results in k-fold cross validation as in the following table:

Table 8. Support Vector Machine & Naïve Bayes Average Value Results

Naive Bayes				Support Vector Machine			
Accuracy	Precision	Recall	F1-score	Accuracy	Precision	Recall	F1-score
72%	68%	68%	68%	73%	75%	64%	64%

With the data above, it can be seen that the accuracy and precision of SVM is superior between one and seven percent to naïve Bayes, while in terms of recall and f1-score, naïve Bayes is superior to Support Vector Machine (SVM) by four percent. In comparison, it was found that the majority of data in the Genshin Impact game review tended to be negative, as in the chart below:

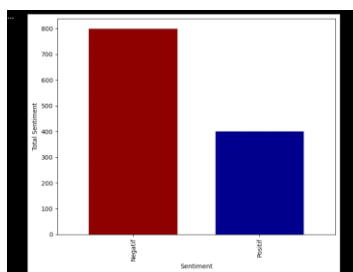


Figure 23. SVM Graphic

4. CONCLUSION

This research shows that the biggest share of Genshin Impact user reviews is negative, while the SVM algorithm (Support Vector Machine) is more effective than Naïve Bayes algorithm in sentiment classification, having a higher accuracy (73%) and precision (75%). However, the inaccuracies of the study, including a small dataset, language bias (Indonesian), and the use of only two algorithms (Naïve Bayes and SVM) without more advanced models like Random Forest or BERT, should be noted. Further research has to deal with additional platforms, languages, and types of sentiment beyond positive and negative. The work has direct relevance to the development of the sectors using sentiment analysis, for example gaming, e-commerce, and social media, which highpoints the productivity of SVM in data processing and its impact on user experience.

ACKNOWLEDGEMENTS

By expressing thanks to the presence of God Almighty for all the blessings that have been given so far, in the end the author can complete this assignment well. The author presents articles on this Undergraduate Program in the form of a simple book. The title of the article, which the author took is as follows, "Comparison of Naïve Bayes and Support Vector Machine Methods in Sentiment

Analysis of Genshin Impact Game Reviews". And all the parties who are too many to mention one by one so that this writing can be realized. The author realizes that the writing of this article is still far from perfect, therefore the author requests constructive criticism and suggestions for the perfection of writing in the future.

REFERENCES

- Abualigah, L., Alfar, HE, Shehab, M., & Hussein, AMA (2020). Sentiment Analysis in Healthcare: A Brief Review. *Studies in Computational Intelligence*, 874, 129–141. https://doi.org/10.1007/978-3-030-34614-0_7
- Acheampong, F.A., Wenyu, C., & Nunoo-Mensah, H. (2020). Text-based emotion detection: Advances, challenges, and opportunities. *Engineering Reports*, 2(7), 1–24. <https://doi.org/10.1002/eng2.12189>
- Anwar, K. (2022). KLIK: Scientific Study of Informatics and Computer Analysis of Sentiment of Instagram Users in Indonesia on Smartphone Reviews Using Naive Bayes. *KLIK: Scientific Study of Informatics and Computer*, 2(4), 148–155. <https://djournals.com/klik>
- Asrumi, Suharijadi, D., Setiari, AD, & Wulanda, DP (2023). SENTIMENT ANALYSIS AND OPINION EXCAVATION. English: Eureka Media Aksara.
- Chang, X., Zhen, T., & Wang, Y. (2023). Exploring the Innovative Pathways of Chinese Traditional Culture's Foreign Communication through "Domestic Games Going Abroad": A Case Study of Genshin Impact. *International Journal of Education and Humanities*, 8(1), 147–150. <https://doi.org/10.54097/ijeh.v8i1.7253>
- Darmawan, G., Alam, S., & Sulistyono, MI (2023). Sentiment Analysis Based on User Reviews of the Mypertamina Application on Google Playstore Using the Naive Bayes Method. *STORAGE – Scientific Journal of Engineering and Computer Science*, 2(3), 100–108. Data Mining and Algorithm Applications.
- Handayani, K., Lisnawanty, L., Latif, A., Firdaus, MR, & Hasan, FN (2021). Comparison of C4.5 and Naive Bayes Algorithms in Determining Blood Donor Eligibility Status. *Sistemasi*, 10(3), 676. <https://doi.org/10.32520/stmsi.v10i3.1440>
- Husada, HC, & Paramita, AS (2021). Sentiment Analysis of Airlines on the Twitter Platform Using the Support Vector Machine (SVM) Algorithm. *Teknika*, 10(1), 18–26. <https://doi.org/10.34148/teknika.v10i1.311>
- Husada, IN, & Toba, H. (2020). The Effect of Class Balancing Method on the Accuracy Level of Sentiment Analysis on Indonesian Tweets. *Journal of Informatics Engineering and Information Systems*, 6(2), 400–413. <https://doi.org/10.28932/jutisi.v6i2.2743>
- Iskandar, JW, & Nataliani, Y. (2021). Comparison of Naive Bayes, SVM, and k-NN for Aspect-Based Gadget Sentiment Analysis. *RESTI Journal (System Engineering and Information Technology)*, 5(6), 1120–1126. <https://doi.org/10.29207/resti.v5i6.3588>
- Jagdale, R. S., Shirsat, V. S., & Deshmukh, S. N. (2019). Sentiment analysis on product reviews using machine learning techniques. In *Advances in Intelligent Systems and Computing* (Vol. 768). Springer Singapore. https://doi.org/10.1007/978-981-13-0617-4_61
- Jimmy Alga, Cindi Wulandari, BI (2024). Sentiment Analysis of Youtube Applications on Google Play Store Using Machine Learning. *RESOLUTION: Informatics and Information Engineering - ISSN 2745-7966*, 6(1), 197–207.
- Kasim, Anita Ahmad, MS (2019). Support Vector Machine (SVM) Algorithm for Economic Classification of Population Recipients of Government Assistance in Simpang Raya District, Central Sulawesi. *APTİKOM National Seminar*, 568–573.
- King, D.L., Delfabbro, P.H., Billieux, J., & Potenza, M.N. (2020). Problematic online gaming and the COVID-19 pandemic. *Journal of Behavioral Addictions*, 9(2), 184–186. <https://doi.org/10.1556/2006.2020.00016>
- Kusnadi, R., Yusuf, Y., Andriantony, A., Ardian Yaputra, R., & Caintan, M. (2021). Sentiment Analysis of the Genshin Impact Game Using Bert. *Rabit: Journal of Technology and Information Systems Univrab*, 6(2), 122–129. <https://doi.org/10.36341/rabit.v6i2.1765>
- Maulidah, N., Supriyadi, R., Utami, DY, Hasan, FN, Fauzi, A., & Christian, A. (2021). Prediction of Diabetes Mellitus Disease Using Support Vector Machine and Naive Bayes Methods. *Indonesian Journal on Software Engineering (IJSE)*, 7(1), 63–68. <https://doi.org/10.31294/ijse.v7i1.10279>
- Nazir, A., Rao, Y., Wu, L., & Sun, L. (2022). Issues and Challenges of Aspect-based Sentiment Analysis: A Comprehensive Survey. *IEEE Transactions on Affective Computing*, 13(2), 845–863. <https://doi.org/10.1109/TAFFC.2020.2970399>
- Noor Hasan, F. (2024). Analysis of CapCut Application User Sentiment on Play Store Reviews Using the Naive Bayes Method. *Media Online*, 4(4), 2272–2280. <https://doi.org/10.30865/klik.v4i4.1555>

- Ph.D. Ummul Aiman, SPDKASHMA Ciq. MJMP, Suryadin Hasda, MPZF, M.Kes. Meilida Eka Sari, MPMKNA (2022). *Quantitative Research Methodology*.
- MP Nanda Saputra (Ed.), Muhammad Zaini Publishing Foundation. Muhammad Zaini Publishing Foundation Member of IKAPI. Pravina, AM, Cholissodin, I., & Adikara, PP (2019). *Sentiment Analysis of Airline Opinions on Twitter Documents Using Support Vector Machine (SVM) Algorithm* (Vol. 3, Issue 3).<http://j-ptiik.ub.ac.id>
- Soraya, F. A., & Indriyanti, A. D. (2024). Comparison of the Naïve Bayes Algorithm and Support Vector Machine in Sentiment Analysis of "Teman Bus" Application. *Journal of Emerging Information System and Business Intelligence (JEISBI)*, 5(2), 118-125.
- Mathapati, P. M., Shahapurkar, A. S., & Hanabaratti, K. D. (2017). Sentiment analysis using Naïve Bayes algorithm. *International Journal of Computational Science and Engineering*, 5(7), 75-77.
- Anreaja, L. J., Harefa, N. N., Negara, J. G. P., Priyantara, V. N. H., & Prasetyo, A. B. (2022). Naive Bayes and Support Vector Machine Algorithm for Sentiment Analysis Opensea Mobile Application Users in Indonesia. *JISA (Jurnal Informatika dan Sains)*, 5(1), 62-68.
- Asriguna, C. I. M., Pudjiantoro, T. H., Sabrina, P. N., & Hadiana, A. I. (2022). Sentiment Analysis of Live. on Digital Provider Application Using Naive Bayes Classifier Method.
- Admojo, F. T., Risnanto, S., Windiawati, A. W., Innuddin, M., & Mualfah, D. (2024). Comparison of Naïve Bayes and Random Forest Algorithm in Webtoon Application Sentiment Analysis. *Innovation in Research of Informatics (INNOVATICS)*, 6(1).
- Butsianto, S., Fauziah, S., Naya, C., & Maulana, F. (2024). Sentiment Analysis Of Indosat's Mobile Operator Services On Twitter Using The Naïve Bayes Algorithm. *Brilliance: Research of Artificial Intelligence*, 4(1), 245-254.
- Hermanto, H., Kuryanti, S. J., & Khasanah, S. N. (2019). Comparison of Naïve Bayes Algorithm, C4. 5 and Random Forest for Classification in Determining Sentiment for Ojek Online Service. *Sinkron: jurnal dan penelitian teknik informatika*, 3(2), 266-274.