Optimized support vector machine for sentiment analysis of game reviews

Bryan Leonardo Supriyatna, Farica Perdana Putri
Department of Informatics, Faculty of Engineering and Informatics, Universitas Multimedia Nusantara, Tangerang, Indonesia

Article Info

Article history:

Received Jan 8, 2024
Revised May 11, 2024
Accepted Jun 18, 2024

Keywords:

Game review
Particle swarm optimization
Sentiment analysis
Support vector machine

ABSTRACT

The rapid development of games has made game categories diverse, so there are many opinions about games that have been released. Sentiment analysis on game reviews is needed to attract potential players. Sentiment analysis is carried out using the support vector machine (SVM) and particle swarm optimization (PSO) algorithms. SVM training was conducted with a linear kernel, the 'C' value parameter was 10 resulting in an accuracy value of 97.28%. The SVM algorithm optimized using the PSO method produces an accuracy of 97.61% using the parameters c1 is 0.2, c2 is 0.5 and w is 0.6. Based on these results, sentiment analysis using PSO-based SVM optimization has been successfully carried out with an increase in accuracy of 0.33%. This game review has a sentiment value from neutral to positive so this game can be recommended to other players.

Corresponding Author:

Farica Perdana Putri
Department of Informatics, Faculty of Engineering and Informatics, Universitas Multimedia Nusantara Tangerang, Indonesia
Email: farica@umn.ac.id

1. INTRODUCTION

In this day and age where people are looking for forms of entertainment, many are turning to gaming to relieve the daily grind. The pandemic period is also an important factor in the surge in the number of gamers. Recently, users of Steam, which is the largest game buying and selling portal for computer users to play games officially, broke a new record of reaching 30 million people who opened the application in 2022 Steam according to the imagine games network (IGN) website [1].

The rapid development of games has led to a variety of game categories, so there are many opinions on games that have been released. Players will look at reviews first before trying to play the game, so that the time they have spent is not wasted. The price of games that continues to soar until now can reach 69.99 US dollars according to the Kotaku website [2] in 2022. The price also affects the purchase of a game. There needs to be something that can help determine whether the game is feasible and in accordance with the interests of the player. Therefore, a rating system is needed that can review the experiences of other players who have played the game to find out whether the game can be recommended or not. Reviews on games are very useful in helping players choose which game to buy, this is evidenced by the interaction of positive and negative reviews affecting 81% of players [3].

Sentiment analysis or opinion mining is the computational study of people's opinions, sentiments, emotions, and attitudes towards an entity such as a product, service, issue, event, topic, and its attributes [4]. Thus, sentiment analysis allows tracking the public mood about a particular entity to create actionable knowledge [5], [6]. By allowing users to actively engage in defining the product requirements through their input, end users may be able to contribute valuable insights into requirements for specific products, which could be advantageous to product owners and engineers [7]. Sentiment analysis can be done with several methods, but the most popular ones are Naive Bayes (NB), random forest (RF), K-nearest neighbors (KNN), decistion tree (DT), logistic regression (LR), and support vector machine (SVM). SVM is a popular classifier because of its ability to deliver higher generalization performance when the input features space has a high-dimension embedding [8]. The basic idea behind SVM is to categorize information separately using hyperplanes to maximize the margin between them.

Many studies have used machine learning approaches to detect the sentiment of a game. A study by [9] used SVM to classify sentiment on game reviews originating from the Steam online platform. The results obtained the highest accuracy reached 97%. Sentiment analysis on game reviews to evaluate video game acceptance was conducted by [10] using Portuguese Bralizilian language. Several classifiers are used in detecting sentiment, namely LR, SVM, and RF. Based on the experiment results, it was found that SVM managed to get the best performances for all four metrics: accuracy, precision, recall, and F1-score, with an accuracy of 82.54%. Meanwhile, LR and RF get the second and third positions, respectively. LR produces an average accuracy of 82.40% and RF of 79.89%. A comparative study on sentiment analysis on game reviews was conducted by Tan *et al.* [11]. Several machine learning algorithms are used, including SVM, multi-layer perceptron (MLP), extreme gradient boosting (XGB), LR, and multinomial Naïve Bayes (MNB). In comparison to the five algorithms, SVM has 91% accuracy because SVM performs classification based on hyperplanes rather than probabilities which is more suitable for text classification with a large number of features. SVM also produced competitive accuracy with MNB and deep neural network (DNN) in [12] in classifying sentiment in game reviews. Term frequency and inverse document frequency (TF-IDF) and bag of words (BoW) are used as document representation for SVM and MNB, while deep averaging network (DAN) and transformer are used for DNN. Evolutionary algorithms (EAs) are a type of optimization algorithm inspired by natural selection and Darwinian survival of the fittest. They are intended to solve optimization and search issues by simulating natural selection, genetic recombination, and mutation. EAs have been widely applied to various optimization problems, such as feature selection, function optimization, and parameter tuning in machine learning. Several EAs utilized to optimized the SVM are genetic algorithm (GA) [13], particle swarm optimization (PSO) [14], and ant colony optimization (ACO) [15]. The PSO methodology allows for numerous optimization methods, including raising the attribute weight of all attributes or variables used, selecting attributes, and selecting features. This study investigated the effects of PSO for feature selection to optimize SVM in game review sentiment analysis, while all the aforementioned studies focus on word representation and SVM to classify the sentiment.

2. RESEARCH METHOD

The methodology of this study consists of several steps and is presented in this section. The steps include data description and collection, data preparation, and the proposed method. Each of these steps is described in detail.

2.1. Data description and collection

The data utilized in this study is a review of Baldur's Gate 3 after patch 5, dated November 30, 2020. Data collection was done using a dataset of reviews from the Steam digital store, as it represents the majority of computer players. The data was collected on Kaggle [16]. Total data divided into 253,976 positive reviews and 7,079 negative reviews. An example of the review data used is described in Table 1. The columns utilized in training are review and voted_up, while timestamps_created, written_during_early_access, and received_for_free are used as parameters to filter the data. voted_up will be True if the sentiment is positive and false if it is negative. timestamps_created is used to filter review data after patch 5. written_during_early_access indicates that the review was written when the version of the game was still a beta version and not the official version of the game, therefore the review is only used when written_during_early_access is False. received_for_free indicates that the game is a game that was obtained for free which is likely a game from a sponsor or gift from the publisher. Therefore, to eliminate bias from reviews, we only use data when received_for_free equals False.

Table 1. Sample of game reviews

Recommendationid	Review	Timestamp_created	Voted_up	Written_during_early_access	Received_for_free
153560814	Game hit right mark	1702542971	True	False	False
153560623	Took like hour understand basic	1702542657	True	False	False
153560414	Game play stori first turn base rpg game love far	1702542275	True	False	False
153560343	Gale babi girl	1702542158	True	False	False
153559963	Yeswithout f n doubt	1702541518	True	False	False

2.2. Data preparation

Data preparation consists of three phases. These phases include data pre-processing, vectorization using TF-IDF, and data splitting. Figure 1 illustrates this process.

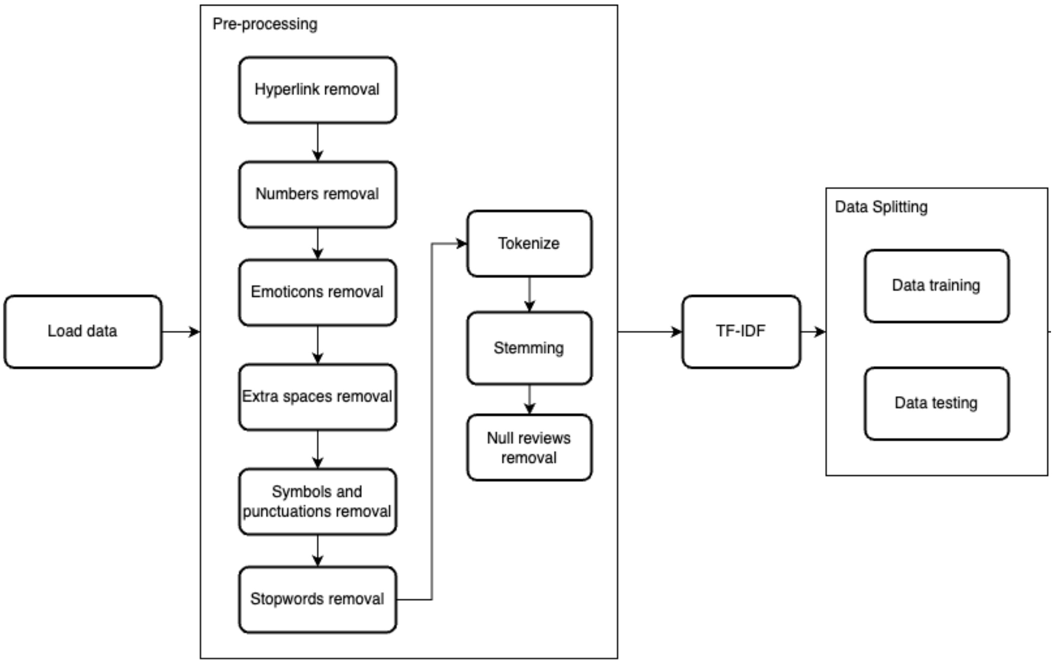


Figure 1. The flow of data preparation

2.2.1. Pre-processing

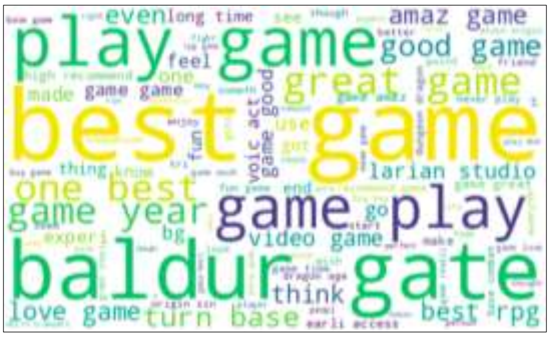
After the review data is filtered by removing early access reviews, text pre-processing is performed to clean the data. First, the text review is converted into lowercase, then not informative things such as hyperlink, numbers, emoticons, extra spaces, punctuations are removed. Stopwords or common words that do not provide information, for example, conjunctions (*and*, *or*) and nouns (*is*, *were*, *was*), are also removed. Then, the review sentences are also separated word-by-word into tokens or it is called tokenization. In addition to facilitating prediction, this phase is also used for the stemming process, where the affixed words will be converted into their root words.1.

The result of stemming is unique word set which is expected to improve the accuracy of the model [17]. The empty data after pre-processing is converted to the empty string (“ ”) and the empty string is removed. Table 2 shows the result of text reviews before and after the pre-processing step.

Table 2. The results of pre-processing step

Text review before pre-processing	Text review after pre-processing
This game hits all the right marks. 10/10	Game hit right mark
Took me like 11 hours to understand the basics	Took like hour understand basic
10/10 game play and story! It’s my first turn based rpg game, and I have been Loving it so far: D	Game play stori first turn base rpg game Love far
Gale is so baby girl	Gale babi girl
YES, \n\nWITHOUT A F****N DOUBT.	Yeswithout f n doubt

The review data is also depicted using a word cloud, which aims to provide a big picture of positive and negative reviews. Figure 2 shows the word cloud of the review data. Figure 2(a) is a word cloud of positive review words and it can be seen that the examples of words that appear most often are “best game”, “baldur gate”, “great game” and “one best”. Figure 2(b) is the word cloud of negative reviews and some of the words that appear frequently are “combat”, “game”, “bug” and “charact”.



(a)

(b)

Figure 2. Word cloud of the review data (a) word cloud of positive reviews and
(b) word cloud of negative reviews

2.2.2. TF-IDF

Feature extraction is done to retrieve important review data. Feature extraction is done with the TF-IDF method where each word that appears is weighted, the reviews will also be converted into vectors using the 'ngram_range' parameter. An n-gram is a contiguous sequence of n items drawn from a particular sample of text or audio. In this context of TF-IDF vectorizer, an n-gram is a sequence of words. Here, we used ngram_range between 1 to 3, it specifies that unigrams, bigrams, and trigrams will be considered when generating TF-IDF features. TF-IDF is one of the methods for term or word weighting. Specifically, it is used to extract core words (i.e., keywords) from documents, calculate the degree of similarity between documents, determine search rankings, etc. TF (term frequency) means the occurrence of certain words in documents. Words with a high TF value have an important meaning in the document. The TF value can be calculated by (1) and tf_t is the number of occurrences of the term *t*.

$$tf_t = 1 + \log tf_t \tag{1}$$

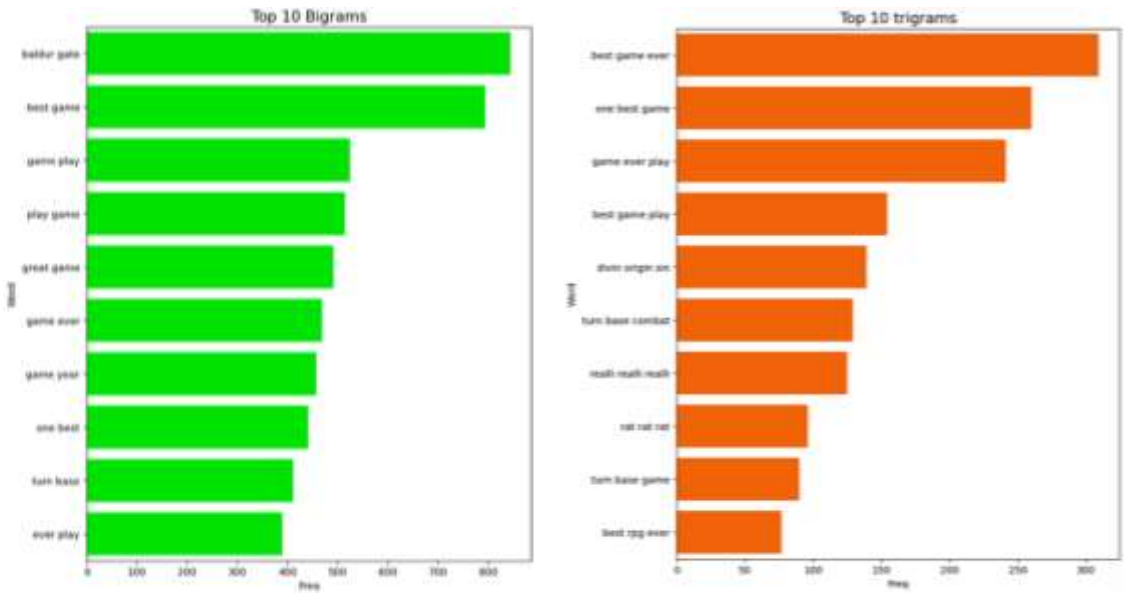
DF (document frequency) implies the number of times a particular word appears in a set of documents. It counts the occurrence of a word in multiple documents, not just in a single document. IDF (inverse document frequency), the inverse of DF, is used to assess the importance of terms in all documents. A high IDF value means that rare words are found across documents, thus increasing their importance [18]. The DF value can be calculated by (2). *D* is the number of documents and df_t is the number of documents available is the term *t*.

$$idf_t = \log\left(\frac{D}{df_t}\right) \tag{2}$$

After the calculation of TF and IDF is done, we can calculate the TF-IDF value using (3), where the value of $W_{t,d}$is the weight of term *t* in document *d*.

$$W_{t,d} = tf_t \times idf_t \tag{3}$$

The process of feature extraction is used to get crucial review data. The TF-IDF approach is used for feature extraction; each word that appears is given a weight, and the output is transformed into a vector. Reviews can be represented using TF-IDF by making a graph of the most frequently occurring words as shown in Figure 3. The bigrams of the most frequently occurring terms is displayed in Figure 3(a). "Best game" has more than 700 occurrences, and "baldur gate" has 800 occurrences. The top ten trigram terms that occur frequently are displayed in Figure 3(b). "One best game" and "best game ever" are two of the most often used phrases, with over 200 and 300 instances, respectively.



(a) (b)

Figure 3. Top 10 of the most frequently n-grams (a) most frequently occurring bigrams and (b) most frequently occurring trigrams

2.2.3. Data splitting

The whole data in the dataset splitted into data training and data testing with percentage of 80% and 20%, respectively. The total sample for training data is 208,844, and the total sample for testing data is 52,211. Table 3 describes the examples of positive and negative reviews from the dataset. The review data was filtered by removing early access reviews, which are reviews when the game is on trial.

Table 3. Example of positive and negative reviews

Positive reviews	Negative reviews
Goty imo	Bad writing and oversimplified combat
Best RPG that came out in years	kinda mid tbh witcher 3 had better vibes
Best dnd base game ever, just be prepared to die alot	Game crashed and made me verify files twice
Just an incredibly good game. The devs perfected and distilled their previous work on the Divinity games and this manages to capture the spirit of the original Baldur's Gate games as well. So polished, love the scope, etc. Just an amazing game.	The game is an unbearably buggy mess. I have encountered so many bugs it isunreal. I have a had enemies hit me with melee attacks from far beyond where theyshould be able to and then to cast healing word, a touch spell, on my ally I hadto move all the way to where they were. I have also encountered bugs that havestopped me from saving in both singleplayer and multiplayer resulting in progresslost on multiple occasions. The game could be good but wait until its not a buggy shit show anymore.

2.3. Proposed algorithm

This sub-section describes the method proposed in this study and the steps taken to obtain sentiment classification results. After the data is vectorized using TF-IDF and split into training and testing data, the training data will be forwarded to PSO for feature selection. The features referred to here are tokens because the data used is text data. Later, the selected features by PSO are used for training SVM in classifying sentiment.

2.3.1. PSO

PSO is a simple optimization method to modify several parameters. PSO converges fast and has few parameters to alter, therefore the computing time of this technique is also reduced. Since several particles attempt to find a solution, the probability of becoming trapped in an ideal local solution is reduced [19]. Initially, particles are placed in positions using (4) and (5) and then perform a search for the optimal value of a particular objective function through exploration and exploitation. The fitness value of the objective function at that position is also stored, which is calculated by (6) [20].

$$x_0^i = x_{min} + rand(0,1)\,(x_{max} - x_{min}) \tag{4}$$

$$v_0^i = v_{min} + rand(0,1)\,(v_{max} - v_{min}) \tag{5}$$

$$F_{k+1}^i = f\left(X_{k+1}^i\right) \tag{6}$$

where,
- x = particle's position
- v = particle's velocity.
- i = particle index.
- $f(x)$ is the objective function

Each particle will have a pbest (personal best) and a *gbest* (global best) value. This pbest value is the best particle position during the iteration performed ($F_{k+1}^i < F_k^i$), while gbest is the particle position value that is closest to the target ($F_{k+1}^{b_1} < F_k^b$). The movement of particles in a flock depends on three factors, namely pbest, gbest, and velocity [21]. The particle velocity formula can be calculated using (7).

$$v_{k+1}^i = wv_k^i + c_1\, rand\left(pbest_i - x_k^i\right) + c_2\, rand\left(gbest_i - x_k^i\right) \tag{7}$$

Where,
- $pbest_i$ = personal best particle i
- w = inertia weight (usually between 0.9-0.4)
- $gbest_i$ = global best particle i
- c_1 = personal learning factor (usually between 0-1)
- c_2 = global learning factor (usually between 0-1)

First, the parameter values c_1 and c_2 must be initialized. Next, use (5) to determine the velocity, and (6) to assess the fitness value. Keep record of the pbest and gbest values. If the requirements aren't satisfied, use (7) to assess the particle velocity and (8) to update the particle position. The fitness value is updated continually during iteration to ensure that the gbest and pbest values satisfy the requirements.

$$x_{k+1}^i = x_k^i + v_{k+1}^i \tag{8}$$

2.3.2. SVM

SVM is a straightforward and adaptable machine learning technique that may be used to address a variety of categorization issues. Particularly, SVM produces balanced predicted performance, even in research with small sample sizes [22]. Using cluster data as a starting point, the SVM approach creates a line of separation (decision boundary) or hyperplane line from the cluster data. Maximum likelihood lines that lie in a space and classify data separated by non-linear or linear boundaries can be constructed by finding a set of hyperplanes that separate two or more classes of data points. After the construction of hyperplanes, SVM finds the distance between the input classes, and the input elements on the hyperplane are called support vectors. From a given set of training samples labeled positive or negative, the hyperplane divides the positive or negative training samples so that the distance between the margin and the hyperplane is maximized. If there is no hyperplane that can divide positive or negative samples, SVM will choose a hyperplane that divides the samples as closely as possible while still maximizing the distance to the closest example that is strictly split [23]. Different classes are predicted based on the data points and the side on which they lie on the hyperplane, which serves as the decision boundary in SVM [24]. The kernel function plays a crucial role in transforming the input data into a higher-dimensional space where the data becomes linearly separable, including linear, radial basic function (RBF), and Polynomial kernel.

Linear kerneal is the simplest kernel, without using the gamma value (γ), using (9). x_i is the value of the training data, x_j is the value of the test data, and $k(x_i.x_j)$ is the kernel value.

$$k(x_i.x_j) = x_i^T x_j \tag{9}$$

RBF is a non-linear kernel, using the gamma value parameter (γ>0) as a determinant of the flexibility of this kernel, can be described by (10). This kernel is suitable for data that cannot be solved linearly with a high level of accuracy and precision [20].

$$k(x_i.x_j) = exp\left(-\gamma.\left\|x_i.x_j\right\|^2\right) \tag{10}$$

Polynomial kernel is a non-linear kernel, using the parameter value of the gamma value (γ>0) and the value of d as the coefficient of the penalty degree for flexibility as described in (11).

$$k(x_i.x_j) = \gamma\left(x_i^T.x_j + r\right)^d \tag{11}$$

A large gamma value will also calculate training data that is far from the decision boundary but will cause a small accuracy value, and the value of C is a free parameter, the value of r is a bias, and the gamma and r parameters have a strong relationship [25].

3. RESULTS AND DISCUSSION

In this section, the experiment results are described and analyzed. We also explain the model parameters used in SVM training, which then become the basis for tuning hyperparameters in the optimized-SVM model using PSO, which in this article will be referred to as SVM-PSO. A discussion about sentiment analysis evaluation on Baldur’s Gate 3 based on the reviews is also presented.

3.1. Experimental setup

This study uses a 3.1 GHz Intel Core i5-12500H processor with memory: 16 GB RAM. Google Colab and libraries such as scikit-learn, seaborn, and pandas are used to execute Python code and data visualization. The proposed approach was executed using PSO iterations of 5 and the number of particles of 30. The hyperparameters tuning for SVM included kernel, gamma, and C.

3.2. Evaluation measures

Evaluation in this study uses a confusion matrix where precision, accuracy, recall, and F1-score values will be sought to determine the suitability of the algorithm that has been used. Accuracy measures the overall correctness of the model’s predictions, and is calculated as the ratio of correctly predicted instances, to the total number of instances in the dataset based on (12).

$$Accuracy = \frac{True\ Positive + True\ Negative}{True\ Positive + True\ Negative + False\ Negative + False\ Positive} \tag{12}$$

The F1-score is the harmonic mean of precision and recall, resulting in a single metric that balances both measurements, as shown in (13). Precision is the proportion of true positive (TP) predictions among all positive predictions made by the model, as shown in (14). Recall is defined as the proportion of true positive (TP) predictions among all real positive cases in the dataset, as shown in (15).

$$F1 - score = \frac{2 \times Precision \times Recall}{Precision + Recall} \tag{13}$$

$$Precision = \frac{True\ Positive}{True\ Positive + False\ Positive} \tag{14}$$

$$Recall = \frac{True\ Positive}{True\ Positive + False\ Negative} \tag{15}$$

3.3. Baseline model

A grid search is performed in order to find the best kernel and parameters of SVM. The tested hyperparameters are described in Table 4. After doing a grid search, the best parameter is found to be a linear kernel with a ‘C’ value of 10. The evaluation results for the negative class show a 19% F1-score, 11% recall, and 89% precision. The precision, recall, and F1-score values for the positive class are 97%, 100%, and 99%, respectively. The accuracy value is 97.28%.

Table 4. SVM parameters

Kernel	C	Gamma
Linear	1, 10	-
RBF	1, 10	1, 0.1
Sigmoid	1, 10	1, 0.1

3.4. SVM-PSO results

While other research merely highlights the performance of SVM without any optimization, this study focuses on the impact of the evolutionary PSO method on the outcomes of sentiment categorization using SVM. The test scenario was carried out by changing the PSO parameters c_1 and c_2 with values of 0.2 and 0.5 and parameter w with values of 0.4, 0.6, and 0.9. Based on our experiments, we found that using PSO as feature selection affects the performance of SVM. This is evidenced by the increased accuracy and F1-score of SVM-PSO compared to SVM, which can be seen in Table 5. The best PSO parameters are c_1=0.2, c_2=0.5, and w=0.6, with the highest accuracy value. This could be because the particle movement is more constant when the c1 and c2 parameter values are the same; the w value has the most impact on the particle movement. The particles move continuously when the c_1 and c_2 parameters have different values. In the meantime, the particle movement will be dominated by the pbest value if the value of parameter c_1 is greater and the gbest value if the value of parameter c_2 is greater. In case the w value is too high, the particle will undergo excessively early migration.

In addition, it can be seen that the Recall and F1-score values for the negative class are very low compared to the positive class. This is due to the difference in the amount of data for each class, the positive class has much more data than the negative class. Therefore, further studies are needed on how to mitigate imbalanced classes and their effect on the performance of SVM-PSO.

Table 5. Results of SVM-PSO and SVM

c_1	c_2	w	Accuracy	Precision		Recall		F1-score	
				-1	1	-1	1	-1	1
0.2	0.2	0.4	97.36	100	97	12	100	22	99
0.2	0.2	0.6	97.45	87	98	18	100	29	99
0.2	0.2	0.9	97.40	100	97	14	100	24	99
0.2	**0.5**	**0.6**	**97.61**	**94**	**98**	**22**	**100**	**35**	**99**
0.5	0.2	0.6	97.53	100	98	18	100	30	99
	SVM		97.28	89	97	11	100	19	99

3.5. Sentiment analysis

Evaluation of sentiment analysis carried out using textblob produces sentiment values, which can be seen in Figure 4. Based on this figure, it can be seen that the reviews have many sentiment values that lean towards neutral and positive, namely -0.5 to 1 or neutral to positive with a count of 3500 reviews. Figure 5 is a comparison chart of reviews included in the "recommendation" with a polarity value of -0.5 to 1.0, with the most reviews on neutral around 3,500, while reviews that include "no recommendation" have a polarity value of -0.75 to 0 or negative to neutral as much as around 200. With reviews that have neutral and positive sentiment values, this game can be recommended to other players.

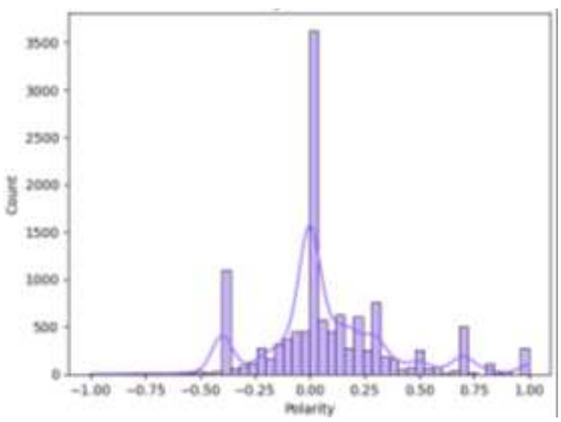


Figure 4. Polarity distribution of data reviews

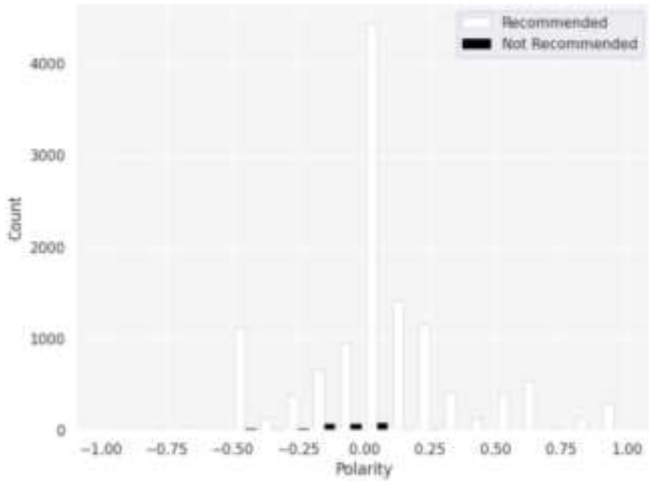


Figure 5. Sentiment polarity distribution of reviews based on recommendation

3. CONCLUSION

The SVM method has been implemented in game sentiment analysis, with the best parameter being a linear kernel with c being 10. The experimental results show that the negative class has a precision of 89%, a recall of 11%, and a F1-score of 19%. The positive class has a precision value of 97%, a recall value of 100%, and an F1-score value of 99%. The accuracy value is 97.28%. SVM optimization with the PSO algorithm is carried out using different parameter variations. The highest results are with values of c_1=0.2, c_2=0.5, and w=0.6, and the accuracy is 97.61%. The evaluation value for the negative class is a precision of 94%, a recall of 22%, and a f1-score of 45%. Meanwhile, the positive class has a precision of 98%, a recall of 100%, and a f1-score of 99%. SVM-PSO has a higher accuracy of 0.33% compared to the SVM base model. With a sentiment rating ranging from neutral to positive, this game can be recommended to other players. The application of word embedding, such as word2vec or BERT, and strategies for managing the imbalanced dataset between positive and negative classes would be taken into consideration for further research.

REFERENCES

[1] A. Bankhurst, "Steam breaks its own concurrent record as 30 million users were online at one time this weekend - ign southeast asia," *IGN Southeast Asia*, 2022. https://sea.ign.com/steam-deck/191728/news/steam-breaks-its-own-concurrent-record-as-30-million-users-were-online-at-one-time-this-weekend.

[2] Z. Zwiezen, "The era of $70 games truly begins this fall," *kotaku*, 2022. https://kotaku.com/70-dollar-games-60-ubisoft-ea-gotham-knights-ps5-xbox-1849593604 (accessed Sep. 28, 2022).

[3] G. Andreev, D. Saxena, and J. K. Verma, "Impact of review sentiment and magnitude on customers' recommendations for video games," in *2021 International Conference on Computational Performance Evaluation, ComPE 2021*, Dec. 2021, pp. 992–995, doi: 10.1109/ComPE53109.2021.9752380.

[4] B. Liu, *Sentiment analysis: Mining opinions, sentiments, and emotions*. Cambridge University Press, 2015.

[5] A. Ligthart, C. Catal, and B. Tekinerdogan, "Systematic reviews in sentiment analysis: a tertiary study," *Artificial Intelligence Review*, vol. 54, no. 7, pp. 4997–5053, Oct. 2021, doi: 10.1007/s10462-021-09973-3.

[6] L. Mostafa, "Student sentiment analysis using gamification for education context," in *Advances in Intelligent Systems and Computing*, vol. 1058, 2020, pp. 329–339.

[7] R. Setiabudi, N. M. S. Iswari, and A. Rusli, "Enhancing text classification performance by preprocessing misspelled words in Indonesian language," *Telkomnika (Telecommunication Computing Electronics and Control)*, vol. 19, no. 4, pp. 1234–1241, Aug. 2021, doi: 10.12928/TELKOMNIKA.v19i4.20369.

[8] B. AlBadani, R. Shi, and J. Dong, "A novel machine learning approach for sentiment analysis on Twitter incorporating the universal language model fine-tuning and SVM," *Applied System Innovation*, vol. 5, no. 1, p. 13, Jan. 2022, doi: 10.3390/asi5010013.

[9] I. M. Urriza and M. A. A. Clarino, "Aspect-based sentiment analysis of user created game reviews," in *2021 24th Conference of the Oriental COCOSDA International Committee for the Co-Ordination and Standardisation of Speech Databases and Assessment Techniques, O-COCOSDA 2021*, Nov. 2021, pp. 76–81, doi: 10.1109/O-COCOSDA202152914.2021.9660559.

[10] Y. HaCohen-Kerner, D. Miller, and Y. Yigal, "The influence of preprocessing on text classification using a bag-of-words representation," *PLoS ONE*, vol. 15, no. 5, p. e0232525, May 2020, doi: 10.1371/journal.pone.0232525.

[11] J. Y. Tan, A. S. K. Chow, and C. W. Tan, "A comparative study of machine learning algorithms for sentiment analysis of game reviews," *The Journal of The Institution of Engineers, Malaysia*, vol. 82, no. 3, Nov. 2022, doi: 10.54552/v82i3.101.

[12] S. Ruseti, M. D. Sirbu, M. A. Calin, M. Dascalu, S. Trausan-Matu, and G. Militaru, "Comprehensive exploration of game reviews extraction and opinion mining using nlp techniques," in *Advances in Intelligent Systems and Computing*, vol. 1041, 2020, pp. 323–331.

[13] P. D. Windha Mega and Haryoko, "Optimization of parameter support vector machine (SVM) using genetic algorithm to review go-jek's services," in *2019 4th International Conference on Information Technology, Information Systems and Electrical Engineering, ICITISEE 2019*, Nov. 2019, pp. 301–304, doi: 10.1109/ICITISEE48480.2019.9003894.

[14] R. Obiedat *et al.*, "Sentiment analysis of customers' reviews using a hybrid evolutionary SVM-based approach in an imbalanced data distribution," *IEEE Access*, vol. 10, pp. 22260–22273, 2022, doi: 10.1109/ACCESS.2022.3149482.

[15] V. Malik, R. Mittal, J. Singh, V. Rattan, and A. Mittal, "Feature selection optimization using ACO to improve the classification performance of web log data," in *Proceedings of the 8th International Conference on Signal Processing and Integrated Networks, SPIN 2021*, Aug. 2021, pp. 671–675, doi: 10.1109/SPIN52536.2021.9566126.

[16] H. Yafie, "Baldur's gate 3 steam reviews," *kaggle*, 2024. https://www.kaggle.com/datasets/harisyafie/baldurs-gate-3-steam-reviews.

[17] D. Gunawan, F. P. Putri, and H. Meidia, "Bershca: bringing chatbot into hotel industry in Indonesia," *Telkomnika (Telecommunication Computing Electronics and Control)*, vol. 18, no. 2, pp. 839–845, Apr. 2020, doi: 10.12928/TELKOMNIKA.V18I2.14841.

[18] S. W. Kim and J. M. Gil, "Research paper classification systems based on TF-IDF and LDA schemes," *Human-centric Computing and Information Sciences*, vol. 9, no. 1, p. 30, Dec. 2019, doi: 10.1186/s13673-019-0192-7.

[19] S. Malakar, S. Sen, S. Romanov, D. Kaplun, and R. Sarkar, "Role of transfer functions in PSO to select diagnostic attributes for chronic disease prediction: An experimental study," *Journal of King Saud University - Computer and Information Sciences*, vol. 35, no. 9, p. 101757, Oct. 2023, doi: 10.1016/j.jksuci.2023.101757.

[20] D. K. Choubey, S. Tripathi, P. Kumar, V. Shukla, and V. K. Dhandhania, "Classification of Diabetes by Kernel based SVM with PSO," *Recent Advances in Computer Science and Communications*, vol. 14, no. 4, pp. 1242–1255, Jul. 2019, doi: 10.2174/2213275912666190716094836.

[21] D. Saputra, W. S. Dharmawan, and W. Irmayani, "Performance comparison of the SVM and SVM-PSO algorithms for heart disease prediction," *International Journal of Advances in Data and Information Systems*, vol. 3, no. 2, pp. 74–86, Nov. 2022, doi: 10.25008/ijadis.v3i2.1243.

[22] D. A. Pisner and D. M. Schnyer, "Support vector machine," in *Machine Learning: Methods and Applications to Brain Disorders*, Elsevier, 2019, pp. 101–121.

[23] J. Cervantes, F. Garcia-Lamont, L. Rodríguez-Mazahua, and A. Lopez, "A comprehensive survey on support vector machine classification: Applications, challenges and trends," *Neurocomputing*, vol. 408, pp. 189–215, Sep. 2020, doi: 10.1016/j.neucom.2019.10.118.

[24] D. K. Dake and E. Gyimah, "Using sentiment analysis to evaluate qualitative students' responses," *Education and Information Technologies*, vol. 28, no. 4, pp. 4629–4647, Apr. 2023, doi: 10.1007/s10639-022-11349-1.

[25] K. V. Kamran, B. Feizizadeh, B. Khorrami, and Y. Ebadi, "A comparative approach of support vector machine kernel functions for GIS-based landslide susceptibility mapping," *Applied Geomatics*, vol. 13, no. 4, pp. 837–851, Dec. 2021, doi: 10.1007/s12518-021-00393-0.

BIOGRAPHIES OF AUTHORS

Bryan Leonardo Supriyatna was born in Tangerang, August 20, 1999, graduated from his studies at Multimedia Nusantara University in 2023 with a major in Bachelor of Informatics. He is interested in database administration, front-end website development, android development and machine learning. He has interned as a website developer for W3O Indo Scientia. He can be contacted at email: leonardos.bryan@gmail.com.

Farica Perdana Putri was born on January 31, 1993 in Indonesia. She received a bachelor's degree in computer science from Universitas Multimedia Nusantara in 2014 and an M.Sc. in computer science from National Taiwan University in 2017. She currently works as a lecturer at Universitas Multimedia Nusantara and is pursuing her Ph.D. at UCSI University. Her research interests mainly focus in semantic analysis on natural language processing (NLP), artificial intelligence, computer vision, machine learning, and related fields, where she has successfully been granted some research grants from the government. She can be contacted at email: farica@umn.ac.id.