IJCSIT
International Journal of Computer Science & Information Technology
ISSN: XXXX-XXXX / Volume 1 Issue 1 January-March 2025 / Page No: 29-33
Paper Id: IJCSIT-V1I1P105

Original Article

Sentiment Analysis Using SVM Classifier in Data Mining: A Machine Learning Approach

Chetna Khaparde
Advisor ERP Functional Analyst, Gainwell Technologies LLC, USA.

Received Date: 15 January 2025 Revised Date: 01 February 2025 Accepted Date: 27 February 2025

Abstract: *An vital part of data mining and natural language processing, abstract sentiment analysis offers understanding of public opinion, customer happiness, and social trends. This work uses a machine learning framework to explore the sentiment classification tasks' application of the Support Vector Machine (SVM) classifier. On a dataset of 10,000 labeled product reviews, a strategy comprising text preprocessing, TF-IDF feature extraction, and SVM model training is used in an organized fashion. With increased accuracy and resilience in categorizing positive, negative, and neutral emotions, the SVM classifier shows better performance than Naive Bayes and logistic regression models. Common in text mining applications, high-dimensional and sparse data management is confirmed by the results to be effective using SVM. The usefulness of SVM in real-time sentiment analysis applications is underlined in this research also together with ideas for improving performance with hybrid or ensemble techniques. investigates the use of Support Vector Machine (SVM) classifiers in sentiment analysis, a main activity in natural language processing (NLP) and data mining. The purpose is to assess SVM's ability to categorize text data into positive, negative, or neutral attitudes. Preprocessing a dataset of text reviews, TF-IDF feature extraction, SVM model training, and performance evaluation of the SVM model follow the study. SVM's efficiency is validated by a comparison analysis including different classifiers.*

Keywords: *Sentiment Analysis; Support Vector Machines; SVM; Data Mining; Machine Learning; Text Classification; Opinion Mining; TF-IDF; Natural Language Processing; Product Reviews.*

I. INTRODUCTION

Sentiment analysis is a method used to recognize and classify thoughts conveyed in a work of text. Accurate and scalable sentiment analysis systems have become much more in demand as online material explodes through reviews, blogs, and social media. It is absolutely important in disciplines including political forecasting, brand monitoring, customer feedback analysis, and corporate intelligence. Rule-based systems have long been in use, but more complex and flexible models have taken front stage as machine learning has emerged. Among these, Support Vector Machine (SVM) has become well-known for its great accuracy—even in high-dimensional, sophisticated data common of text data. Appropriate for both binary and multiclass sentiment classification applications, SVM is a supervised learning model seeking an ideal hyperplane to partition classes. This work aims to classify product review opinions as favorable, negative, or neutral using SVM. Data collecting, cleaning, TF-IDF vectorizing preprocessing, and standard classification metrics model training and evaluation comprise the approach. Additionally shown to show SVM's effectiveness is a comparison performance study including logistic regression and Naive Bayes. Ten thousand tagged reviews make up the used dataset, which offers a strong basis for evaluation. By means of our work, we hope to underline the significance of SVM in real-world applications and its pragmatic benefits for sentiment analysis chores. Furthermore included in this work are possible improvements using ensemble techniques and integration with deep learning to solve present constraints in sentiment recognition including domain adaptability, context sensitivity, and sarcasm. entails figuring the emotional tone of text material. Opinion mining, product reviews, and social media monitoring all benefit from it extensively. Sentiment classification's accuracy has improved thanks to machine learning; SVM's resilience in managing high-dimensional data has made it a potent tool. This work addresses binary and multiclass sentiment categorization using SVM.

II. LITERATURE REVIEW

Previous studies showing the value of machine learning techniques in sentiment analysis—which has changed dramatically within the past two decades Early research mostly depending on lexicon-based methods and rule-based systems. These methods, meantime, sometimes lacked scalability and flexibility to enter new fields. Classifiers as Naive Bayes, decision trees, and logistic regression—which greatly enhanced the capacity to categorize emotions from vast volumes of unstructured text—were adopted when machine learning emerged. While logistic regression provides strong performance on linearly separable datasets, naive bayes has been generally acknowledged for its simplicity and computing economy. By contrast, especially for high-dimensional data like textual information, Support Vector Machines (SVMs) have routinely

surpassed these models in terms of accuracy and generalization. Because of its margin maximizing method, which improves model robustness, several research have underlined SVMs' strengths in sentiment analysis. SVMs can reach exceptional accuracy in movie and product review classification, according to studies like Pang and Lee (2002). Moreover, convolutional and recurrent neural networks as well as other neural network-based models with great capacity for deep feature extraction have become rather popular recently. Still, these models may call for big datasets and substantial processing capacity. SVM is therefore appropriate for a broad spectrum of uses since it offers a decent trade-off between accuracy and economy. More recently, hybrid methods—that is, SVM combined with feature selection methods or ensemble learning—have been investigated to maximize performance. The literature shows a clear trend of SVM outperforming conventional models in text classification, therefore proving its applicability in sentiment analysis, particularly in sparse data and limited training examples.

III. METHODOLOGY

This work uses a structured machine learning pipeline running Support Vector Machine (SVM) classifier sentiment analysis. Starting with the choice and preparation of the dataset, the approach comprises several phases. From a publicly available dataset, 10,000 tagged product reviews were acquired; labels were classified as favorable, negative, or neutral. To standardize the language, the raw text data went through thorough preparation including tokenizing, stop-word elimination, and lemmatization. Term Frequency-Inverse Document Frequency (TF-IDF) was used to extract features that convert the text into numerical feature vectors denoting the corpus term importance. The SVM classifier was trained using these vectors then as input. The SVM model was selected for its capacity to efficiently divide classes in high-dimensional space and generalize with low overfitting. Using an 80/20 train-test split, the model was trained then hyperparameter tuned via cross-valuation and grid search. The performance of the classifier was assessed using key assessment criteria including accuracy, precision, recall, and F1-score. To evaluate relative efficiency, these measures were then matched with those from logistic regression models and naive bayes. The approach also uses a pie chart to visualize sentiment distribution in order to grasp dataset class balance. Implementing tools included Python, scikit-learn, and pandas; effort was taken to guarantee repeatability and openness all through the process. This method guarantees a comprehensive study and accurate results, therefore guiding the deployment of SVM-based sentiment analysis models in practical contexts.

The 10,000 tagged product reviews utilized in this study come from a public repository and guarantee accessibility and openness by means of their source. Every review falls into either good, negative, or neutral one of three sentiment categories. The database offers a fair picture of customer opinions by including a wide spectrum of product categories like electronics, home appliances, clothes, and literature. From small sentences to multi-sentence stories, the reviews range in length and complexity and provide a rich corpus for classifier training. The dataset was examined and changed where needed to guarantee class balance, therefore preserving an almost 45% positive, 35% negative, and 20% neutral sentiment distribution. This evenly distributed supports objective model evaluation and training. Additionally cleansed during the preprocessing stage was the dataset for noise—duplicate entries, pointless content, and incorrect labeling. Though not directly employed in sentiment classification to keep emphasis on the textual content, metadata including product ID, category, review date, and reviewer rating was also available. To guarantee labeling quality, a portion of the data was personally checked; any variances were corrected. This well chosen dataset provides a strong basis for developing a strong sentiment analysis model with the SVM classifier. Used in this investigation are 10,000 labeled product reviews taken from a public repository. Every review comes under either positive, negative, or neutral labels.

Any text-based machine learning pipeline must first go through preprocessing, which is particularly important for sentiment analysis projects since it directly affects the classifier's performance. Initially tokenizing the raw text data, each review is broken up into distinct words or tokens. This stage helps to enable the later textual input filtering and normalizing. Stop-word removal—that is, the elimination of often used terms as "the, "is,„ and "in—which usually do not convey significant sentiment information—follows tokenization. Lemmatizing then helps to simplify words to their base or dictionary form, so uniting variants of the same word (e.g., "running, ran, and runs" become "run"). To guarantee uniformity over the dataset, extra preprocessing procedures include contractions handling, lowercasing of all text, and punctuation removal. Following text cleaning, feature extraction is performed using the Term Frequency-Inverse Document Frequency (TF-IDF) method, which computes the relative importance of every word in the context of the whole corpus so converting the cleaned text into numerical vectors. Transposing unstructured text into a format fit for SVM classifier input requires this stage. To minimise noise and lower dimensionality, care is also taken to exclude too rare or too frequent terms. Python tools like NLTK and Scikit-learn help to ensure both efficiency and repeatability by means of the preprocessing pipeline. These preprocessing chores not only get the data ready for modeling but also greatly increase the accuracy of the model by guaranteeing that only pertinent, standardized, high-quality input gets to the classifier.

SVM finds the hyperplane in a high-dimensional space that best divides the classes. It translates the input characteristics into this space and maximizes the margin between several sentiment classes for sentiment analysis. Because of its capacity to effectively manage high-dimensional, sparse data—qualities frequent in text datasets resulting from the nature of language—SVM has great strength. Using kernel functions such linear, polyn, or radial basis function (RBF), the classifier converts the input data into a higher-dimensional space where the ideal separation hyperplane can be discovered. Based on early testing, this work chose a linear kernel that offered the optimum trade-off between computing complexity and accuracy. Using TF-IDF feature vectors, the SVM was trained; grid search and 5-fold cross-valuation helped to improve hyperparameters including the regularizing parameter (C) and kernel type. The Scikit-learn library offers effective tools for SVM training and prediction, hence the classifier was used there. Through weight assignment to various variables depending on their relevance in distinguishing classes, the model learns to discriminate between positive, negative, and neutral attitudes during training. SVM has one benefit in that it resists overfitting, particularly in situations when the number of features is far higher than the number of samples—as is typically the case in text classification problems. SVM also offers strong decision boundaries and can handle binary and multiclassification challenges rather successfully. To prove its better performance in sentiment classification, the trained SVM model is tested against baseline models and subjected to standard metrics. Its capacity to generalize effectively on unprocessed data shows the pragmatic value of SVM in real-world sentiment analysis uses. by in a high-dimensional space determining the hyperplane most likely to divide the classes. It translates the input characteristics into this space and maximizes the margin between several sentiment classes for sentiment analysis.

In sentiment analysis, the performance of the SVM classifier depends critically on evaluation measures. They provide a numerical foundation for evaluating the classifier's practical relevance and against other models. By dividing the number of accurate predictions by the total number of predictions, accuracy—which gauges the general correctness of the model's predictions—measures the most often used statistic. In circumstances with imbalanced datasets, accuracy by itself might be deceptive, so further measures are required. Especially crucial when the cost of false positives is significant is the measurement of the percentage of true positive predictions among all the positive predictions generated using precision. Recall, sometimes known as sensitivity, measures the model's capacity to find all pertinent events, therefore capturing the fraction of true positives among all actual positives. When both accuracy and memory are crucial, F1-score offers a fair assessment by harmonic mean of both. These four measures—accuracy, precision, recall, and F1-score—together offer a whole picture of classifier performance. These measures were computed for every sentiment class (positive, negative, neutral) in this work using macro-averaging to compile performance across classes. Confusion matrices were also produced to graphically evaluate class-wide correct and erroneous classification distribution. This study helps to find any prejudices the model might have towards particular groups. Using built-in evaluation tools included in Scikit-learn, metrics were generated to guarantee dependability and consistency. These measures direct hyperparameter tweaking and enable evaluation of preprocessing actions on model accuracy. In the end, evaluation criteria not only confirm the performance of the classifier but also direct iterative development and point up areas needing more optimization.

Classifier	Accuracy	Precision	Recall	F1-Score
SVM	89.5%	90.2%	88.7%	89.4%
Naive Bayes	82.3%	83.0%	81.5%	82.2%
Logistic Regression	85.7%	86.5%	84.2%	85.3

IV. RESULTS

The SVM model was trained and tested using an 80/20 train-test split on a dataset of 10,000 product reviews. It consistently outperformed the benchmark models across all evaluation metrics. In terms of accuracy, the SVM achieved 89.5%, outperforming logistic regression and Naive Bayes, which scored 85.7% and 82.3% respectively. Precision and recall values were particularly strong for the positive sentiment class, demonstrating the model's effectiveness in identifying favorable opinions. The confusion matrix indicated fewer misclassifications for SVM compared to other models, especially in distinguishing between neutral and negative sentiments. The robustness of SVM was further validated through cross-validation, where it maintained a low standard deviation across folds, confirming consistent performance. Additionally, macro-averaged precision, recall, and F1-scores were calculated to account for class imbalance and showed SVM's balanced capability across all sentiment categories. The SVM classifier also exhibited superior generalization on unseen data, indicating minimal overfitting. ROC curves plotted for each sentiment class showed higher area under the curve (AUC) values for SVM, further supporting its discriminative power. Moreover, the training process was optimized using grid search for hyperparameter tuning, with a linear kernel providing the best balance between accuracy and computational cost. Runtime analysis showed that while SVM required more computational resources during training compared to Naive Bayes, it was efficient in prediction time, making it suitable for real-time applications. Visualization of sentiment predictions on a

random sample confirmed that the model could handle linguistic nuances, including polarity shifts and mixed sentiment expressions. These findings collectively highlight SVM's effectiveness, reliability, and practicality in performing sentiment classification tasks in diverse real-world scenarios. model was trained and tested using an 80/20 train-test split. The performance metrics were calculated and compared against Naive Bayes and logistic regression classifiers

V. DISCUSSION

Based on all measured criteria, SVM beats other classifiers, according the results. Its capacity to manage sparse and high-dimensional data qualifies it especially for sentiment analysis projects. Though its training time is rather slower than that of Naive Bayes, the performance increases make its use justified.

VI. CONCLUSION

In conclusion, this work strengthens the importance of Support Vector Machines (SVM) as a strong and efficient classifier for sentiment analysis activities inside the larger framework of data mining and machine learning. For categorizing textual data like product reviews, the model is a recommended choice for high-dimensional and sparse datasets as well as for generalizing to unseen data since it can manage both. Our experiments reveal that SVM not only beats conventional classifiers such as Naive Bayes and logistic regression across many performance criteria—accuracy, precision, recall, and F1-score—but also preserves consistent performance through cross-valuation and real-world simulation. Combining its scalability and adaptability with its better handling of class imbalance, the classifier can be used in e-commerce, customer service, political analysis, and healthcare sentiment tracking among other fields. Moreover, SVM's adaptability for several kernel functions offers versatility in modeling intricate, nonlinear decision boundaries as needed. Although the work used a linear kernel because of its efficiency and efficacy, future research could investigate the effects of alternative kernel types and feature selection techniques to improve performance even more. Especially in complex sentiment settings containing sarcasm, implicit sentiment, or domain-specific language, including SVM into hybrid models or ensemble frameworks may produce even better results. Deep learning methods such word embeddings or neural architectures to improve feature representation would also be a useful direction for advancement. Strong, interpretable, and accurate sentiment categorization tools become more important as online information keeps expanding. Among realistic, well-understood, high-performance models that strike performance-to- computational-efficiency balance is SVM. Reliable sentiment predictions enable SVM to be very helpful for real-time analytics systems and strategic decision-making processes, therefore demonstrating its long-term significance in the changing terrain of natural language processing and data-driven insights.

VII. REFERENCES

[1] Pang, B., Lee, L., then Vaithyanathan. (2002). thumbs up Machine learning methods of sentiment classification. 10, 79–86. ACL-02 Conference on Empirical Methods in Natural Language Processing proceedings

[2] Cortes, C. & Vapnik, V. (1995). Networks of support-vector type. 20(3), 273–297: Machine Learning.

[3] Liu, b. (2012). Opinion mining and sentiment analysis help us. Published by Morgan & Claypool Publishers.

[4] Joachims, T. 1998 is here. Learning with multiple relevant features using support vector machines for text categorization European conference on machine learning, 137–142.

[5] Sevati, F. 2001 In automatic text classification, machine learning Survey of ACM Computing, 34(1), 1–47.

[6] Hovy, E. and Kim, S. M. (2004). figuring out the attitude of people. COLING 2004 proceedings: 1367–1373.

[7] Voll, K., Brooke, J., Tofiloski, M., Taboada, M., & Stede, M. (2011). lexicon-based techniques for sentiment analysis. 37(2), Computational Linguistics: 265–307.

[8] Vapnik, V.N. 1998 (1998). The theory of statistical learning. Wiley-Interscience.

[9] Go, A., R. Bhayani, & L. Huang. 2009 is here. Twitter sentiment categorization under distant supervision. Project Report on CS224N at Stanford.

[10] Zaharakis, I.; Kotsiantis, S. B.; Pintelas, P. (2007). Review of classification methods: Supervised machine learning. Emerging artificial intelligence uses in computer engineering, 160, 3–24.

[11] Salton, G., and Buckley, C. 1988). Term-weighting methods in automatic text search. 24(5) Information Processing and Management, 513–523.

[12] Blei, D. M., Jordan, M. I. and Ng, A. Y. 2003 is the year Latent Dirichlet distribution is 3, 993–1022 Journal of Machine Learning Research.

[13] Kennedy, A.: and Inkpen, D. 2006 is here. Contextual valence shifters help to classify movie reviews in sentiment terms. 22(2), 110–125, computational intelligence.

[14] Turn ey, P. D. 2002 saw Thumbs in favor of or thumbs against? Semantic direction applied to unsupervised review categorization. Notes of the 40th Annual Meeting on Association for Computational Linguistics, 417–424

[15] Wilson, T., Wiebe, J., & Hoffmann, P. (2005) Acknowledging contextual polarity in phrase-level sentiment analysis Notes of HLT/EMNLP, 347–354.

[16] Mullens, T., and Collier, N. (2004). Support vector machine sentiment analysis with several information sources. EMNLP proceedings: 412–424.

[17] Pak, A. & Paroubek, P. 2010 Using Twitter as a corpus for opinion mining and sentiment analysis LREC Proceedings: 1320–1326.

[18] Medhat, W., Korashy, H. & Hassan, A. 2014). Applications and sentiment analysis techniques: a survey Five (4), 1093–1113 Ain Shams Engineering Journal.
[19] Zhang, L.; Wang, S.; & Liu, B. 2018 stands for Sentiment analysis deep learning: a survey Data mining and knowledge discovery: Wiley Interdisciplinary Reviews, 8(4), e1253.
[20] Jiang, L.; Yu, M.; Zhou, M.; Liu, X.; & Zhao, T. 2011 is Twitter sentiment classification with target dependence. ACL-HLT proceedings: 151–160.
[21] Patt, J. 1999 is here. Sequential minimal optimization fast training of support vector machines Kernel Methodological Advances: 185–208.
[22] Witten, I. H., Hall, M. A., Frank, E. 2011 marks. Data mining: pragmatic tools and methods for machine learning. Murray Kaufmann.
[23] Liu, B.; Hu, M.; & Cheng, J. 2005). Opinion observer: Web opinion analysis and comparison study 14th International Conference on World Wide Web proceedings 342–351.
[24] Franklin, R. 2013 (2013). Methodologies and uses for sentiment analysis Notes of the ACM, 56(4), 82–89.
[25] Bird, S.; Klein, E.; Loper, E. In 2009. Python for Natural Language Processing. O'Reilly Media, Inc., here.
[26] Ghosh, S., Chollet, F.; Sengupta, R. 2020 is here. Effective Natural Language Processing in Practice O'tReilly Media.
[27] Barros, R. C.; Basgalupp, M. P.; de Carvalho, A. C.; Freitas, A. A. (2014). an analysis of evolutionary techniques for induction of decision- Trees Systems, 44(3), 362–399 IEEE Transactions on Systems, Man, and Cybernetics.
[28] Wilson, T.; Kouloumpis, E.; Moore, J. 2011. Analysis of Twitter sentiment: The good the bad and the OMG! Fifth International Conference on Weblogs and Social Media (ICWSM) proceedings here.
[29] Sun, A. 2001; Lim, E. P. Hierarchical appraisal and classification of books. IEEE conference on data mining, 521–528.
[30] Jain, A. K.; Duin, R. P. W.; Mao, J. 2000: A review on statistical pattern recognition 22(1), 4–37 IEEE Transactions on Pattern Analysis and Machine Intelligence