

IMPACT OF FEATURE HASHING FOR DIMENSIONAL REDUCTION TECHNIQUE ON MOVIE REVIEW DATASETS

Usman Mahmud^{1*}, Mohammed Hassan², Abubakar Ado³, Abdullahi Abdulwahab⁴, Abdulkadir Abubakar Bichi⁵, Sanusi Abu Darma⁶

¹Al-Qalam University, Katsina. Dept. of Computer Science, Katsina, Nigeria

²Dept. of Software Engr, Bayero University, Kano, Nigeria

³Dept. of Computer Science, Yusuf Maitama Sule University, Kano, Nigeria

⁴Dept. of Computer Science, Bayero University, Kano, Nigeria

⁵Dept. of Computer Science, Yusuf Maitama Sule University, Kano, Nigeria

⁶Dept. of Computer Science, Al-Qalam University, Katsina, Nigeria.

*Corresponding Author

Usman Mahmud

Al-Qalam University, Katsina. Dept. of
Computer Science, Katsina, Nigeria.

Article History

Received: 09.10.2023

Accepted: 19.10.2023

Published: 13.11.2023

Abstract: Sentiment analysis plays a pivotal role in comprehending public sentiment and user feedback. In this study, we investigate the effectiveness of feature hashing, a dimensional reduction technique, in conjunction with classifier enhancement to improve sentiment analysis using a movie review dataset under precision, accuracy and recall evaluation metrics. We specifically address two key aspects: the challenges in the movie dataset that necessitated the application of feature hashing and the significant of the propose approach. Our initial assessment of three baseline classifiers: Linear SVC, K-Nearest Neighbors (KNN), and Bernoulli Naive Bayes (NB) revealed precision scores of 85.92%, 61.57%, and 86.52%, respectively. Recognizing that these classifiers might not fully exploit the dataset's potential due to data redundancy issues, we applied feature hashing (FH) to mitigate these challenges. This reduction technique resulted in a streamlined feature set. Following feature hashing, we observed remarkable precision improvements. Linear SVC achieved 90.60%, KNN demonstrated an impressive 92.43%, and Bernoulli (NB) maintained a high precision level of 90.34%. These outcomes shows the effectiveness of feature hashing in eliminating data redundancy, enabling our classifiers to better capture the underlying sentiment nuances in the movie review dataset. Our proposed approach consistently outperforms existing methods, affirming its capacity to enhance sentiment analysis tasks. These findings underscore the significance of dimensional reduction techniques such as feature hashing in bolstering classifier performance. Our study addresses the unique challenges of the movie dataset and highlights the applicability of these techniques in real-world sentiment analysis scenarios, where accurate sentiment assessment is indispensable for informed decision-making and comprehending user sentiments towards products and services.

Keywords: Sentiment Analysis, Feature Hashing, Data Redundancy, Movie Review Dataset.

1.0 INTRODUCTION

In the digital age, the explosion of data due to powerful computers and widespread internet access has become a norm. This data boom is largely attributed to the prevalence of social media, blogs, emails, and other online platforms that have seamlessly integrated into our daily lives. In response to this data deluge, businesses are finding traditional methods of gathering customer feedback, like surveys, to be inadequate. There is a growing realization that more

sophisticated approaches are needed to gain insights into customer sentiments and preferences. This is where the field of Sentiment Analysis comes into play. Sentiment Analysis, often referred to as SA, is a way of deciphering the emotional tone and opinions expressed in text. It provides a means to gauge how people feel about a particular subject, be it a movie, a product, or an idea. Consider the scenario of analyzing movie reviews; SA allows us to examine reviews as a whole or break them down into smaller components, such as individual sentences or even single words. However, this micro-level analysis poses a challenge. When we

break reviews down into smaller parts, such as individual words or phrases, we end up with an overwhelming amount of information. Imagine the increasing volume of data poses challenges in efficiently managing and analyzing text. Common techniques like "word count vector, TF_IDF, bag of words" and n-gram feature representations for text classification often lead to impractically vast-dimensional data. This can make data mining computationally complex due to the high number of dimensions. To address this, dimensionality reduction methods become essential to improve the performance and reduce the complexity of learning algorithms.

The central concept of dimensionality reduction involves converting the initial high-dimensional data into a lower-dimensional space, making it easier to recognize underlying data patterns. Various models like Latent Dirichlet Allocation (LDA), Latent Semantic Indexing (LSI), Probabilistic Latent Semantic Analysis (PLSA), and Principal Component Analysis (PCA) are commonly employed for dimensionality reduction in text classification.

Nevertheless, when confronted with data of exceptionally high dimensions, often consisting of hundreds of thousands of individual dimensions, the task of transforming data instances into feature vectors in real-time using these models becomes financially burdensome in terms of computational resources. For example, when dealing with a fresh data point, denoted as 'a,' and utilizing a previously acquired LDA model labeled as 'B,' the process of representing 'a' as a feature vector involves the estimation of concealed topics (Z), specifically the posterior probability ($p(Z|a,B)$). In contexts involving high data throughput, such as online text classification, ranking, or ad selection, where the volume of new data points can reach millions, the traditional methods might not be particularly efficient. Consequently, there exists a notable demand for dimensionality reduction techniques that can swiftly transform data into features in real-time, ideally accomplishing this in a consistent timeframe for each feature.

A promising avenue of research in the realm of dimensionality reduction is the utilization of feature hashing. Feature hashing as proposed by (Shi et al. 2009; Weinberger et al. 2009; Forman and Kirshenbaum 2008; Langford, Li, and Strehl 2007) is an approach that presents an affordable and efficient technique for diminishing the feature count. Feature hashing involves introducing random collisions into latent factors. It achieves this by taking the original high-dimensional space and "shrinking" it through the process of feature hashing, employing a hash function to map features to hash keys. The process allowed the multiple features can be assigned (randomly) to the same key, and their counts are "aggregated." While this method is highly effective in reducing the feature count from very high (e.g., 2^{30}) to a more manageable mid-size (e.g., 2^{15}) dimensions, it can lead to a noticeable loss of redundant information, particularly when hash collisions occur among highly frequent features, potentially resulting in significant differences in class distributions which is the key concept of using Feature Hashing on dimensionality reduction.

By considering the impact of Feature Hashing on dimensionality reduction on textual data the research question that this study intended to address is can Feature Hashing serve as an efficient and effective approach for eliminating data redundancy without causing loss of vast information on the content of the dataset when

applied on movie review? Below are the key significant contributions of the study:

1. **Efficiency:** Feature hashing is computationally efficient, making it suitable for handling high-dimensional textual data. It allows for the reduction of feature dimensions with minimal computational cost, which is crucial when dealing with large datasets or real-time processing.
2. **Scalability:** Feature hashing is highly scalable. It can handle large volumes of textual data without a significant increase in processing time. This scalability is essential for applications that involve massive amounts of text, such as web content analysis, social media sentiment analysis, or search engine indexing.
3. **Dimensionality Reduction:** Feature hashing effectively reduces the dimensionality of textual data. By mapping features to a lower-dimensional space, it helps in managing the "curse of dimensionality," which can lead to increased computational complexity and decreased model performance. This dimensionality reduction can lead to faster training and improved model efficiency.

Overall, feature hashing is a valuable technique for reducing dimensionality in textual data, as it combines efficiency, scalability, and effective dimensionality reduction, making it particularly useful in natural language processing and text mining applications.

Related Work

Numerous methods for reducing dimensionality have been explored in the existing body of research, which will be outlined in the following sections. Feature hashing, also known as random clustering, has been explored by researchers like Shi et al. (2009) and Weinberger et al. (2009). They introduced hash kernels as a means to transform high-dimensional input spaces into lower-dimensional ones, demonstrating their effectiveness in large-scale classification and multitask learning when the lower-dimensional space is sufficiently substantial. Additionally, Ganchev and Dredze (2008) provided empirical evidence that hash features yield accurate results in various natural language processing (NLP) applications. Furthermore, Forman and Kirshenbaum (2008) put forward a rapid feature extraction technique that combines parsing and hashing for tasks like text classification and indexing. Caragea & Silvescu, (2009) proposed approach that provides a practical solution to dimensionality reduction for text classification tasks. By combining hashing and abstraction, it helps manage high-dimensional text data efficiently and maintain classification performance. The approach is valuable for applications where large-scale text data classification is essential, such as news article categorization, web content analysis, and scientific document classification. Caragea proposed an approach that focuses on the challenge of handling high-dimensional data in protein sequence classification. It introduces feature hashing as a dimensionality reduction technique and conducts experiments to evaluate its effectiveness in reducing dimensionality while maintaining classification performance. The research aims to provide insights into the applicability of feature hashing in computational biology tasks. Zhao & Xing (2014) conducted a research titled as "HESHING" as a method to address dimensionality challenges in large-scale image categorization with a hierarchical taxonomy structure. It provides theoretical analysis and empirical results to support the effectiveness of this approach in reducing

computational and storage requirements while maintaining classification accuracy. Gao et al., (2016) introduces an approach called Cuckoo Feature Hashing (CCFH) as a novel approach to address the challenges of feature hashing in large-scale sparse feature datasets. CCFH dynamically resolves collisions among predictive features, reduces parameter dimensions, and maintains model sparsity, resulting in improved model performance. Experimental results validate the effectiveness of CCFH in reducing the number of parameters while achieving similar performance. Mahmud et al., (2023) introduces a novel approach to address data redundancy in sentiment classification for movie reviews. It combines Information Gain-based feature selection and Feature Hashing to reduce dimensionality and improve classification accuracy. The experimental results confirm the superiority of the proposed approach over the baseline methods. Ado (2021) propose an innovative feature hashing approach that effectively reduces collisions between features, thus enhancing model performance, especially in binary classification tasks. The proposed method is based on term weight and provides a promising solution for handling high-dimensional sparse feature spaces in large-scale text datasets. Mahmoud & Zanga, (2023) introduces an Information Gain-based feature selection approach for eliminating

un-wanted feature on sentiment analysis, which proves effective in improving the accuracy and precision of sentiment classification models, particularly in the context of movie reviews.

2.0 Material and Method

In this section, we will describe the materials and methods used in the experiments and provide details about the experimental setup.

2.1 Model Framework

It can clearly be seen from Figure 1, the model under goes two stages (i.e training and testing) initially, the document (datasets) undergoes vectorization which will convert all the numerical value into textual form which resultant the feature space. Next step is to apply feature hashing which eliminate and minimize all the redundant feature from datasets which resultant hash space. Next space is to deployed machine learning classifiers for testing the performance of the model which yielded the machine learning prediction as an output. NB the testing stage is hidden from the model and it has a direct access with all process within the model which means that at any steps the model can performing a testing stage from moving the next step.

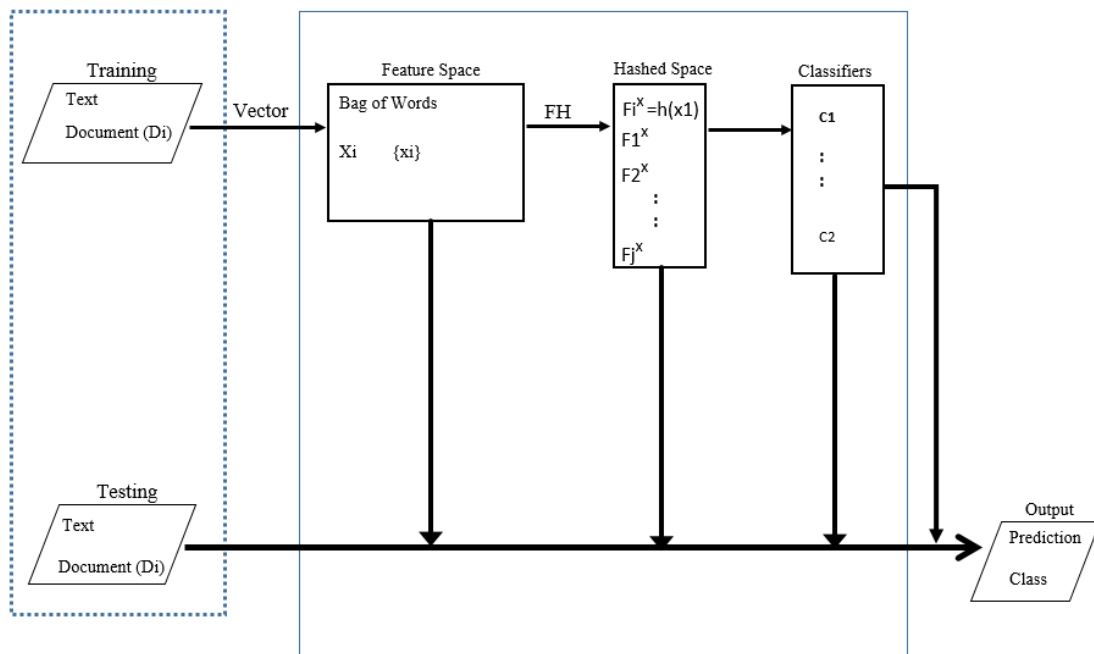


Figure 1 depicts the structural framework of the model.

2.2 Feature Hashing in Movie Review Analysis

Feature hashing, a method for dimensionality reduction, was developed by researchers including Weinberger, Shi, Forman (2008), and Langford (2007). Feature hashing involves the transformation of high-dimensional input vectors y (with dimensionality d) into lower-dimensional feature vectors y^c (with dimensionality c). Within this process, X represents the set of all possible strings, while c and ξ serve as two hash functions. These hash functions, $c: X \rightarrow \{0, \dots, d-1\}$ and $\xi: X \rightarrow \{\pm 1\}$, respectively, work together to map each token in a document into a hash key ($d1$), which represents the token's index in the feature vector y^c . This index is a numerical value ranging from 0 to $d-1$. Each index within y^c stores the value, often denoting "frequency

counts," for the corresponding hash feature. The hash function ξ plays a crucial role by indicating whether to increment or decrement the hash dimension of the token, ensuring that the resulting hash feature vector y^c remains unbiased.

2.3 Dataset and Description

The data used in this study is sourced from the publicly available Stanford dataset, which can be freely accessed on www.kaggle.com. This dataset comprises a collection of 50,000 movie reviews, equally divided into 25,000 positive and 25,000 negative reviews. This balanced distribution of sentiment is a standard practice in sentiment analysis benchmarks. An essential step in applying machine learning algorithms to sentiment analysis is dividing the dataset into training and testing subsets to accurately

evaluate the model's performance. In this instance, the data is split into a 70% portion for training and a 30% portion for testing.

2.4 Classifiers

The performance evaluation of a model when feature hashing is applied to the dataset involves the use of three key classifiers: Bernoulli Naive Bayes, Linear Support Vector Classifier (Linear SVC) and K-Nearest Neighbors (K-NN). These classifiers are widely employed in machine learning and offer unique advantages when dealing with transformed data. Here's an explanation of their significance:

1. **Bernoulli Naive Bayes:** Bernoulli Naive Bayes is a probabilistic classifier well-suited for binary and sparse data. When feature hashing is used, it excels at handling high-dimensional sparse data efficiently. It makes the assumption that features are conditionally independent given the class, which can be a reasonable assumption in certain scenarios (Umarani et al., 2021)
2. **Linear Support Vector Classifier (Linear SVC):** Linear SVC is a robust linear classification algorithm designed to identify the optimal hyper plane that separates data points of different classes. When feature hashing and information gain are applied to the dataset, it becomes possible to significantly reduce the data's dimensionality. This reduction can enhance the efficiency of model training and inference while maintaining or even improving classification accuracy (Mahmoud & Zanga, 2023)
3. **K-Nearest Neighbors (K-NN):** K-NN is a simple yet effective instance-based classification algorithm. It can handle high-dimensional data effectively and is adept at capturing local patterns in the feature space. When the proposed approach is implemented, k-NN can take advantage of reduced dimensionality while still delivering strong performance in classification tasks (Hastie, T. et al., 2009).

In summary, these classifiers, namely Bernoulli Naive Bayes Linear SVC and K-NN, are instrumental in assessing model performance when feature hashing are applied. They can adapt to the characteristics of transformed datasets, efficiently handle high-dimensional or sparse data, and offer solid classification performance.

2.5 Evaluation Metrics

After the preliminary study, data collection (Information relating train-test stages which we have used during the trial stage), data cleaning, data transformation represented with word count vector and data visualization, the results were obtained by being using 3 Machine Learning algorithms, below metrics which we have used, and the results obtained have also been detailed in this section.

After taking the experimental frame work, metrics such as “recall”, “precision “and “accuracy “are selected for evaluating performance of the considered classifier.

2.5.1 Recall

Recall represents the total number of positive classifications out of true class. In our case, it represents the number of articles predicted as true out of the total number of true articles. It can be calculate on below formula:

$$\text{Recall} = \frac{\text{TP}}{\text{TP} + \text{FN}}$$

2.5.2 Precision

Conversely, precision score represents the ratio of true positives to all events predicted as true. In our case, precision shows the number of articles that are marked as true out of all the positively predicted (true) articles it can be computed as:

$$\text{Precision} = \frac{\text{TP}}{\text{TP} + \text{TN}}$$

2.5.3 Accuracy

Accuracy is often the most used metric representing the percentage of correctly predicted Observations, either true or false. To calculate the accuracy of a model performance it can be calculated by using below formula:

$$\text{Accuracy} = \frac{\text{Number of correct Predictions}}{\text{Number of predictions}}$$

$$\text{Accuracy} = \frac{\text{TP} + \text{TN}}{\text{TP} + \text{TN} + \text{FP} + \text{FN}}$$

For binary classification accuracy can be calculated as:

Where FN, FP, TN, and TP show false negative, false positive, true negative, and true positive are defined as follows:

False-negative (FN): Incorrectly predicted negative instances.

False-positive (FP): Incorrectly predicted positive instances.

True negative (TN): Correctly predicted negative instances.

True Positive (TP): Correctly predicted positive instances.

3.0 Result and Discussion

In this section, we present the outcomes of our study, with a particular focus on how our suggested method has impacted sentiment analysis when applied to the Stanford movie dataset. Our investigation is primarily driven by the objective of reducing unnecessary data repetition, eliminating interference from irrelevant information, refining the selection of parameters, and optimizing the extraction of features to achieve more accurate and precise classification results in comparison to the standard methods. Tables 1 provide a concise overview of the comparative performance of three key machine learning classifiers: Linear SVC, K-Nearest Neighbors (KNN), and Bernoulli Naive Bayes. However, for a deeper understanding of the practical implications of these findings, it is imperative to explore the disparities between our proposed approach and the conventional method. Our proposed approach addresses the issue of redundant data through the utilization of Feature Hashing Technique. The improvements observed in classification metrics, such as accuracy and precision, underscore its practical significance. Refer to Table 1 for a detailed breakdown of these results.

Table 1 Illustrate evaluation Metrics on Accuracy, Recall and Precision

Classifier	Accuracy		Recall		Precision	
	All Feature	FS_(FH)	All Feature	FS_(FH)	All Feature	FS_(FH)
Linear SVC	86.12	89.10	86.37	51.86	85.92	90.60
KNN	63.12	88.63	69.77	41.50	61.57	92.43
Bernoulli(NB)	85.41	88.68	83.89	45.89	86.52	90.34

Reference to Table 1 in our study, we applied machine learning algorithms to the Stanford movie dataset, focusing on three fundamental evaluation metrics: accuracy, precision, and recall. The results obtained from both baseline and proposed methods are summarized below:

Accuracy Evaluation Metrics: Initially, we trained three basic classifiers on accuracy evaluation metrics. In the baseline approach, Linear SVC achieved an accuracy of 86.12%, KNN reached 63.12%, and Bernoulli Naive Bayes (NB) scored 85.41%. However, after incorporating the Feature Hashing (FH) dimensional reduction technique to eliminate data redundancy, we observed significant improvements. Linear SVC reached 89.10%, KNN improved to 88.63%, and Bernoulli Naive Bayes (NB) reached 88.68%. These results clearly indicate that our proposed approach outperforms the existing methods in all three cases, demonstrating the practical significance of Feature Hashing.

Precision Evaluation Metrics: In the precision evaluation, the baseline results showed that Linear SVC achieved a precision of 85.92%, KNN had a precision of 61.57%, and Bernoulli Naive Bayes (NB) scored 86.52%. Upon applying Feature Hashing to the dataset, we witnessed remarkable improvements. Linear SVC reached a precision of 90.60%, KNN significantly improved to

92.43%, and Bernoulli Naive Bayes (NB) reached 90.34%. These results underscore the superior performance of our proposed approach, particularly when it comes to precision.

Recall Evaluation Metrics: When assessing recall using the baseline method, Linear SVC achieved a recall of 86.37%, KNN reached 69.77%, and Bernoulli Naive Bayes (NB) scored 83.89%. However, after incorporating Feature Hashing, we observed a substantial decrease in recall for all classifiers. Linear SVC's recall dropped to 51.86%, Bernoulli Naive Bayes (NB) fell to 45.89%, and KNN decreased to 41.50%. This suggests that recall may not be a suitable metric when Feature Hashing is employed.

In summary, our proposed approach, which leverages Feature Hashing for dimensional reduction, consistently outperforms the existing methods in terms of accuracy and precision across all three classifier algorithms. However, it's important to note that Feature Hashing may have a detrimental impact on recall. These findings highlight the practical advantages of our approach in optimizing sentiment analysis on the Stanford movie dataset, particularly in scenarios where accuracy and precision are of paramount importance. Figure 2 shows percentage improvement and differences for all tested classifiers with respect to Feature Hashing (FH) and propose approach in all cases.

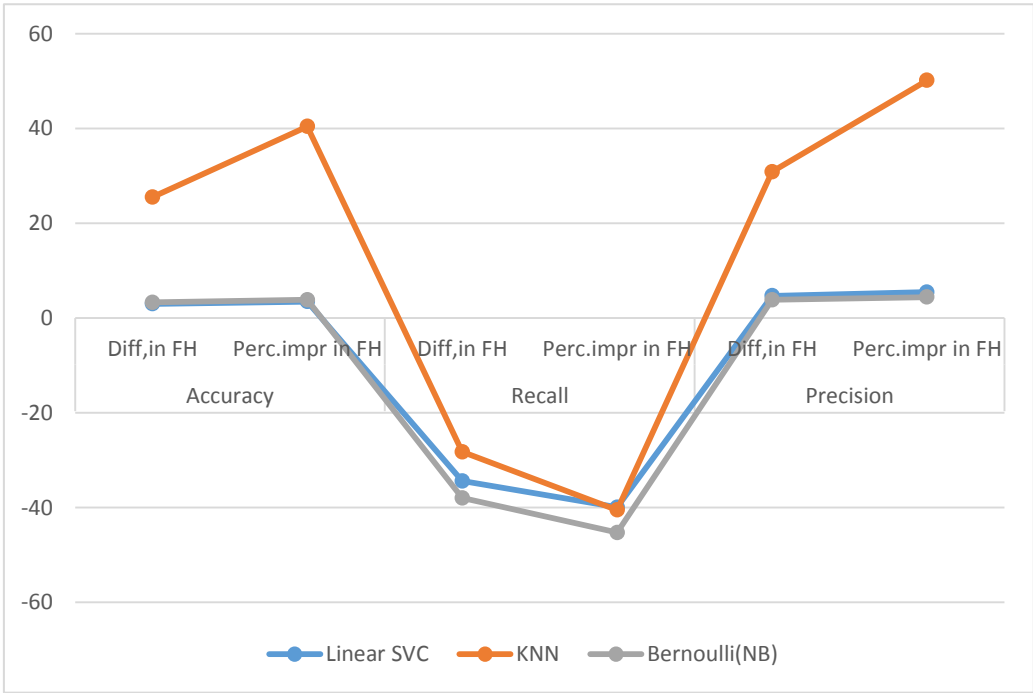


Figure 2 Percentage Improvement & Differences with respect to classifiers on accury, precision and recall metric

Conclusion and Future work

In conclusion, our study focused on improving sentiment analysis using the Stanford movie dataset by applying machine learning algorithms and evaluating them based on accuracy, precision, and recall metrics. Our proposed approach, which involved the use of Feature Hashing for dimensional reduction, yielded significant enhancements in accuracy and precision for all three basic classifiers: Linear SVC, KNN, and Bernoulli Naive Bayes (NB).

Specifically, the results demonstrated that our proposed method outperformed the baseline approaches in terms of accuracy and precision. Linear SVC, KNN, and Bernoulli Naive Bayes (NB) consistently achieved higher accuracy and precision scores after the application of Feature Hashing, highlighting the practical significance of our approach.

However, it's worth noting that the use of Feature Hashing had a notable negative impact on recall, suggesting that this metric may not be suitable when Feature Hashing is applied.

Overall, our findings indicate that our proposed approach, leveraging Feature Hashing, is effective in reducing data redundancy and enhancing sentiment analysis performance. It is particularly valuable when accuracy and precision are crucial in sentiment analysis tasks.

Based on the findings of our study, several promising avenues for future research and work in the field of sentiment analysis and machine learning can be explored:

1. **Optimizing Feature Selection:** Further research can focus on developing more advanced feature selection techniques that not only reduce data redundancy but also maintain or enhance recall performance. Balancing the trade-off between dimensionality reduction and recall is a critical challenge to address.
2. **Hybrid Approaches:** Investigate the potential of combining Feature Hashing with other feature extraction and reduction techniques to achieve a balance between accuracy, precision, and recall. Hybrid methods may provide a more versatile solution for sentiment analysis tasks.
3. **Deeper Evaluation Metrics:** Consider the use of additional evaluation metrics beyond accuracy, precision, and recall. Metrics like F1-score, ROC-AUC, and confusion matrix analysis can provide a more comprehensive understanding of model performance.
4. **Application to Diverse Datasets:** Extend the evaluation of the proposed approach to other sentiment analysis datasets, encompassing various domains and languages, to assess its generalizability and effectiveness in different contexts.
5. **Real-time Sentiment Analysis:** Investigate the feasibility of implementing the proposed method for real-time sentiment analysis, where timely and accurate sentiment predictions are essential, such as in social media monitoring or customer feedback analysis.

By pursuing these avenues for future work, researchers can continue to advance the field of sentiment analysis and develop

more effective, robust, and interpretable machine learning models for a wide range of applications.

References

1. Ado, A. (2021). *A New Feature Hashing Approach Based on Term Weight for Dimensional Reduction*.
2. Caragea, C., & Silvescu, A. (2009). *Combining Hashing and Abstraction in Sparse High Dimensional Feature Spaces*. 3–9.
3. Forman, G., and Kirshenbaum, E. 2008. Extremely fast text feature extraction for classification and indexing. In *Proc. Of the 17th ACM CIKM*, 12211230
4. Ganchev, K., and Dredze, M. 2008. Small statistical models by random feature mixing. In *proceedings of the ACL-2008 Workshop on Mobile Language Processing*.
5. Gao, J., Ooi, C., Shen, Y., & Lee, W. (2016). *Cuckoo Feature Hashing: Dynamic Weight Sharing for Sparse Analytics*. 2135–2141.
6. Hastie, T., Tibshirani, R., & Friedman, J. (2009). *The Elements of Statistical Learning: Data Mining, Inference, and Prediction*. Springer.
7. Langford, J.; Li, and Strehl, A 2007. Vowpal wabbit
8. Mahmoud, U., & Zanga, A. I. (2023). *Application of Feature Selection Dimensionality Reduction Method Based On Information Gain For Sentiment Classification Using Movie Review Dataset*. 2(2017), 1–6.
9. Shi Q., Petterson, J., Dror, G., Langford, J.; Smola, A.; Strehl, A.; and Vishwanathan, V. 2009. Hash kernels. In *Proc. of the Artificial Intelligence and Statistics, Florida 2009*
10. Silvescu, A.; Caragea, C.; and Honavar, V. 2009. Combining super-structuring and abstraction on sequence classification. In *ICDM*, 986–991.
11. Umarani, V., Julian, A., & Deepa, J. (2021). Sentiment Analysis using various Machine Learning and Deep Learning Techniques. *Journal of the Nigerian Society of Physical Sciences*, 3(4), 385–394. <https://doi.org/10.46481/jnsps.2021.308>
12. Weinberger, K., Dasgupta, A., Langford, J., Smola, A., & Attenberg, J. (2009). Feature hashing for large scale multitask learning. *Proceedings of the 26th International Conference On Machine Learning, ICML 2009, Icm1*, 1113–1120.
13. Zhao, B., & Xing, E. P. (2014). *Hierarchical Feature Hashing for Fast Dimensionality Reduction*. <https://doi.org/10.1109/CVPR.2014.263>.