International Journal of Scientific Research in Engineering and Management (IJSREM)
Volume: 07 Issue: 12 | December - 2023 SJIF Rating: 8.176 ISSN: 2582-3930

MultiClass Text Classification Using SupportVector Machine

G. Swetha
CSE(AI&ML)
Malla Reddy University
Hyderabad, India
2111CS020584@mallareddyuniversity.ac.in

G. Thejashwini
CSE(AI&ML)
Malla Reddy University
Hyderabad, India
2111CS020589@mallareddyuniversity.ac.in

V. Tarun Sri
CSE(AI&ML)
Malla Reddy University
Hyderabad, India
2111CS020587@mallareddyuniversity.ac.in

S. Tarun kumar Reddy
CSE(AI&ML)
Malla Reddy University
Hyderabad, India
2111CS020585@mallareddyuniversity.ac.in

J. Tharuni
CSE(AI&ML)
Malla Reddy University
Hyderabad, India
2111CS020588@mallareddyuniversity.ac.in

O. Siddhu
CSE(AI&ML)
Malla Reddy University
Hyderabad, India
2111CS020586@mallareddyuniversity.ac.in

K.Manoj Sagar, M.E.
Assistant Professor AI & ML
School of Engineering
Malla Reddy University, Telangana.
manojsagar2489@gmail.com

1. ABSTRACT

Support vector machine (SVM) was initially designed for binary classification .To solve multi-class problems of support vector machines (SVM) more efficiently, a novel framework, which we call class-incremental learning (CIL) CIL reuses the old models of the classifier and learns only one binary sub-classifier with an additional phase of feature selection when a new class comes. In text classification, where computers sort text documents into categories, keeping up with new information can be tricky. Traditional methods need lots of retraining to adapt. However, Incremental Learning for multi-class Support Vector Machines (SVMs) offers a solution. It lets us update the model with new data while remembering what it learned before .In this project, we'll explore how Incremental Learning makes multi-class SVMs better at handling changing data and even learning about new categories as they appear .There is a problem in addressing the challenge of integrating new classes while maintaining classification accuracy on existing and new classes .The main goal of this project is to create a method that effectively adapts the MC-SVM to evolving data distributions while minimizing the impact on previously learned classes and optimising resource utilization.

Keywords— feature extraction, support vector machine, multi class incremental learning, Gaussian kernel.

2. INTRODUCTION

Text class, a pivotal utility of machine studying, entails categorizing textual content documents into predefined classes. Leveraging the Support Vector Machine (SVM) algorithm, this assignment focuses on classifying articles in the BBC News dataset. This big dataset encompasses articles from distinct classes like business, entertainment, politics, recreation, and technology. The SVM version operates through getting to know styles inside the textual content features extracted from those articles, discerning subtle nuances to predict the respective categories appropriately.

Utilizing this dataset involves an initial section of preprocessing, encompassing steps like tokenization, stemming, and feature extraction, wherein the textual content facts is transformed right into a based numerical layout for SVM processing. The SVM algorithm, famend for its effectiveness in coping with high-dimensional facts, then learns the premiere category barriers to segregate articles into their respective instructions.

The objective of this project lies in now not handiest training an correct SVM version however additionally in deploying it efficiently for real-time classification of latest, unseen articles. Continuous refinement and version optimization are vital, making sure adaptability to evolving linguistic patterns and maintaining high accuracy in classifying news articles into their appropriate classes.

3. LITERATURE REVIEW

Text classification, a fundamental task in natural language processing, involves categorizing textual documents into predefined classes or categories. Support Vector Machines (SVMs) have gained significant attention and application in this field due to their effectiveness in handling high-dimensional data and robustness in

classification tasks. Numerous studies have explored various methodologies to optimize SVMs for text classification. This includes investigating different kernel functions (linear, polynomial, radial basis function) to capture complex relationships within the data. Additionally, researchers have delved into hyperparameter tuning and feature selection techniques to enhance SVM performance. A key aspect of text classification using SVMs involves representing textual data effectively. Traditional bag-of-words (BOW) models and TF-IDF (Term Frequency-Inverse Document Frequency) have been widely used. More recent advancements include word embeddings, such as Word2Vec , which capture semantic relationships among words, enhancing SVMs' understanding of textual context. Challenges in text classification using SVMs include dealing with high-dimensional feature spaces, imbalanced datasets, and handling noisy or sparse text data. Researchers have proposed solutions, such as ensemble methods, active learning, and semi-supervised learning, to improve classification accuracy and address these challenges. Literature often conducts comparative analyses between SVMs and other machine learning algorithms like Naive Bayes, Random Forests, and Neural Networks across different datasets and classification tasks. These studies assess SVMs' strengths in terms of scalability, interpretability, and accuracy compared to alternative methods .SVM-based text classification finds applications across diverse domains including sentiment analysis, document categorization, spam filtering, medical text classification, and social media analysis. Studies demonstrate SVMs' adaptability in handling multiclass problems and their applicability across various textual data types. Recent research focuses on improving the scalability and efficiency of SVMs for large-scale text classification tasks. Distributed computing paradigms and parallelization techniques are explored to optimize computational efficiency while maintaining high classification accuracy.

Few limitations of this project are:
>Computational intensity.
>Sensitivity to Noise .
>Difficulty with large feature spaces.
>Binary classification nature

4. PROBLEM STATEMENT

The objective of this project is to perform multiclass text classification using Support Vector Machines (SVMs) on the BBC News dataset. The primary aim is to develop a robust machine learning model capable of accurately categorizing news articles into predefined classes based on their content.The task involves classifying news articles from the BBC News dataset into distinct categories, including business, entertainment, politics, sport, and tech. Each article in the dataset represents textual information related to these categories.Accurate categorization of news articles is essential for content recommendation systems, information retrieval, and organizing vast amounts of textual data. A robust multiclass text classification model based on SVMs can aid in automating this process, facilitating efficient news categorization for various applications.This project aims to address these research questions and hypotheses by leveraging SVMs to classify news articles accurately, contributing to the advancement of text classification techniques in handling multiclass datasets like the BBC News dataset.The dataset comprises a collection of news articles, where each article belongs to one of the five categories. The articles are in textual format and vary in length, covering diverse topics within each category. This dataset serves as the basis for training and evaluating the SVM-based multiclass text classification model.

To discover what techniques are promising for learning text classifiers, we have to discover more approximately the homes of textual content:
High dimensional input area: When studying textual content classifiers on has to cope with very many (extra than 10000) functions. Since SVMs use overfitting safety which does Text pre-processing Texts are unstructured and use the herbal language of people, which make its semantics difficult for the pc to cope with. So they want essential pre-processing. Text pre-processing particularly segments texts into phrases.

LSA-based totally characteristic extraction and dimensionality discount: LSA is used in this module for the feature extraction and the dimensionality reduction of word-document matrix of training set. Singular values and corresponding singular vectors are extracted via the singular value decomposition of phrase-record matrix, to constitute a new matrix for approximately illustration of the authentic phrase report matrix. Compared with VSM, it could reflect the semantic link between words and the effect of contexts on phrase meanings, cast off the discrepancy of textual content illustration caused by synonyms and polysemes, and decrease the dimension of textual content vectors.

Vectorization of text: In this model, every row vector of the
phrase-record matrix represents a text this is the vectorization
of text. During a trying out process,after each check sample segmented into phrases, the preliminary textual content vectors are mapped to a latent semantic space in this module with the aid of LSA vector space model,
to generate new textual content vectors.

DATA DESCRIPTION

A dataset refers to a structured collection of data points or observations that are organized and stored for analysis or processing. It consists of individual data instances, often arranged in rows (samples, examples, or records) and columns (features or attributes).The dataset we used in this project is bbc news dataset .The dataset comprises news articles gathered from the BBC website. This dataset is collected from Kaggle .Each article belongs to one of five categories: 'business', 'entertainment', 'politics', 'sport', and 'tech'. The goal is to classify these articles into their respective categories using machine learning techniques, particularly Support Vector Machines (SVM).

The dataset's primary purpose is to develop machine learning models for accurately classifying articles into their respective categories. Featuring diverse textual content

from reputable news sources, the dataset offers rich and varied language patterns, enabling the construction of robust classification models. The articles vary in length, complexity, and writing style, presenting an authentic representation of real-world news articles. This dataset's balanced distribution across different categories ensures fair representation and minimizes biases, contributing to effective model training and evaluation. The research focuses on developing a robust machine learning framework employing SVMs to classify news articles from the BBC News dataset into distinct categories, enhancing the efficiency of information retrieval and categorization.By leveraging machine learning techniques, particularly SVMs, this research endeavors to create an accurate and scalable model for categorizing news articles across multiple domains, utilizing the rich textual data available in the BBC News dataset.

5. METHODOLOGY

SVM is an effective technique for classifying high dimensional data. Unlike the nearest neighbour classifier, SVM learns the optimal hyper plane that separates training examples from different classes by maximizing the classification margin. It is also applicable to data sets with nonlinear decision surfaces by employing a technique known as the kernel trick, which projects the input data to a higher dimensional feature space, where a linear separating hyperplane can be found. SVM avoids the costly similarity computation in high-dimensional feature space by using a surrogate kernel function. It is known that support vector machines (SVM) are capable of effectively processing feature vectors of some 10 000 dimensions, given that these are sparse. Several authors have shown, that support vector machines provide a fast and effective means for learning text classifyers from examples. Documents of a given topic could be identifyed with high accuracy Support Vector Machine (SVM) is supervised learning method for classification to find out the linear separating hyperplane which maximize the margin, i.e., the optimal separating hyperplane (OSH) and maximizes the margin between the two data sets. An optimal SVM algorithm via multiple optimal strategies is developed in presented latest technique for documents classification. Among all the classification techniques SVM and Naïve Bayes has been recognized as one of the most effective and widely used text classification methods provide a comprehensive comparison of supervised machine learning methods for text classification. One remarkable property of SVMs is that their ability to learn can be independent of the dimensionality of the feature space. SVMs measure the complexity of hypotheses based on the margin with which they separate the data, not the number of features. This means that we can generalize even in the presence of many features, if our data is separable with a wide margin using functions from the hypothesis space. The same margin argument also suggests a heuristic for selecting good parameter settings for the learner (like the kernel width in an RBF network). The best parameter setting is the one which produces the hypothesis with the lowest VC Dimension. This allows fully automatic parameter tuning without expensive cross-validation.

Why Should SVMs Work Well for Text Categorization?
To find out what methods are promising for learning text classifiers, we should find out more about the properties of text.

High dimensional input space: When learning text classifiers, one has to deal with very many (more than 10000) features. Since SVMs use overfitting protection, which does not necessarily depend on the number of features, they have the potential to handle these large feature spaces.

Few irrelevant features: One way to avoid these high dimensional input spaces is to assume that most of the features are irrelevant. Feature selection tries to determine these irrelevant features. Unfortunately, in text categorization there are only very few irrelevant features. All features are ranked according to their (binary) information gain. Then a naive Bayes classifier is trained using only those features ranked 1- 200, 201- 500, 501-1000, 1001-2000, 2001-4000, 4001-9962. A classifier using only that worst" feature has a performance much better than random. Since it seems unlikely that all those features are completely redundant, this leads to the conjecture that a good classifier should combine many features (learn a \dense" concept) and that aggressive feature selection may result in a loss of information.

Document vectors are sparse: For each document, the corresponding document vector contains only few entries which are not zero.

MODEL ARCHITECTURE

The model starts by collecting a bunch of news articles from the BBC dataset, where each article covers various topics like business, entertainment, politics, sport, and tech.Before understanding the articles, the model cleans them up by removing unnecessary stuff like punctuation and sorts the words to make them easier to analyze.The model then looks at each article and counts how many times different words appear, making a list of word counts for each article. It's like figuring out which words appear the most in each piece of news.The model uses Support Vector Machines (SVMs) to draw lines (not real lines, but mathematical ones) between different groups of articles. It's like drawing boundaries between business, entertainment, politics, sport, and tech articles based on their word counts. the model quickly looks at the words in it, checks which side of the lines they fall on, and decides which category (business, entertainment, etc.) the article belongs to based on the boundaries it drew earlier. The dataset is divided into training and testing sets for model development and evaluation. The Support Vector Machine (SVM) algorithm is employed for classification, learning to assign articles to their respective categories based on extracted features. Model performance is assessed using metrics like accuracy, precision, recall, and F1-score Once trained and validated, the SVM model is ready for deployment to categorize new articles in real-time.
Continuous monitoring of the model's performance allows for iterative improvements, including potential retraining with updated data to adapt to evolving patterns in news articles. This architecture provides a structured approach for effective text classification using SVM on the BBC News dataset.

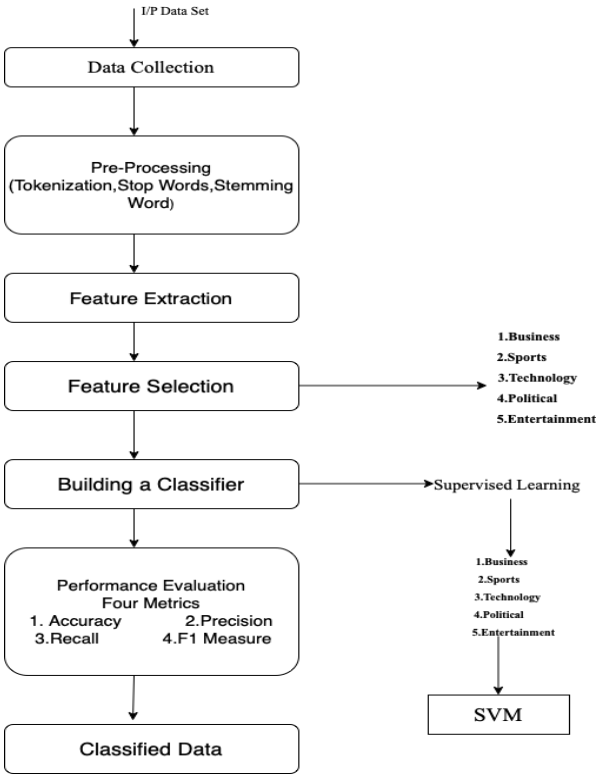

Fig1: Model architecture

DATA PREPROCESSING TECHNIQUES

Application of preprocessing methods can improve the dataset's quality for Text classification tasks. There is a widespread variety of text preprocessing methods.

>Lower casing : Converting all text to lowercase to ensure uniformity and avoid duplication due to case differences.

>Tokenization :Breaking text into tokens/words: Splitting text into individual words or phrases (tokens) to analyze them separately.

>Stemming/Lemmatization: Reducing words to their base/root form, Transforming words to their base form to reduce redundancy Stemming: Truncating words to their root form using heuristic algorithms.

Lemmatization: Mapping words to their base form using lexical knowledge.

>TF-IDF Vectorization: Term Frequency-Inverse Document Frequency means Assigning weights to words based on their frequency in a document and across the corpus.

> Addressing missing data: Dealing with missing text values through imputation or removal, ensuring the integrity of the dataset.

6. EXPERIMENTAL RESULTS

Accuracy measures the overall correctness of the model's predictions. It provides a general overview of how often the model correctly classified articles into their respective categories. However, in the context of imbalanced datasets, accuracy might not be the sole indicator of model performance. Precision indicates the accuracy of the model when it predicts a specific category. High precision is crucial in scenarios where false positives carry significant consequences. Recall measures the model's ability to capture all instances of a particular category. High recall is essential when missing relevant articles is undesirable. The F1-score is the harmonic mean of precision and recall. It balances precision and recall, providing a single metric for overall model performance. It is particularly useful when there is a need for a balanced approach between precision and recall. The confusion matrix is a tabular representation of true positive, true negative, false positive, and false negative predictions. It offers a detailed breakdown of the model's performance for each category, aiding in error analysis and identifying patterns of misclassifications.

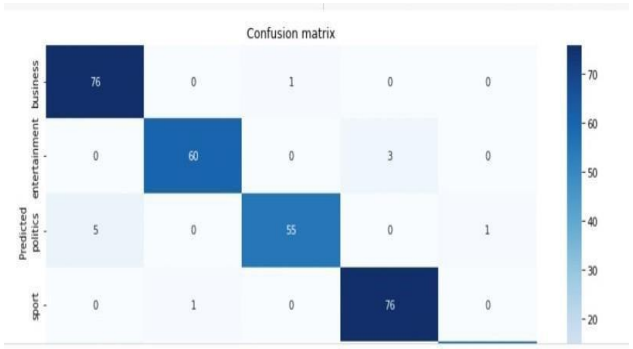

Fig 2: Confusion matrix

These are instances where the model correctly predicted a positive class. In the context of text classification, it represents articles correctly categorized into their respective topics. Instances where the model correctly predicted a negative class. In text classification, this corresponds to articles accurately identified as not belonging to a particular category. These are instances where the model incorrectly predicted a positive class. In text classification, it represents articles wrongly classified as belonging to a certain category. Instances where the model incorrectly predicted a negative class. In text classification, this refers to articles that should have been assigned to a category but were missed. The classification report summarizes precision, recall, and F1- score for each category. It provides a comprehensive evaluation of the model's performance across different categories, aiding in understanding strengths and weaknesses. Hyperparameter tuning using RandomizedSearchCV influenced the values of key evaluation metrics. The process identified optimal parameters, contributing to the model's overall performance and effectiveness in text classification. The model was tested on real-world data extracted from specified URLs, representing diverse categories. The predictions aligned well with the true categories, showcasing the model's practical utility beyond the training dataset. Differences in accuracy between the training and testing sets were analyzed. The goal was to assess whether the model exhibited signs of overfitting or underfitting and how well it generalized to unseen data. Visualizations were used to depict trends in evaluation metrics during different project phases. Patterns observed in these visual representations provided insights into the model's performance evolution.

Classification report:

	precision	recall	f1-score	support
0	0.92	0.99	0.95	77
1	0.98	0.95	0.97	63
2	0.98	0.90	0.94	61
3	0.96	0.99	0.97	77
4	0.98	0.96	0.97	56
accuracy			0.96	334
macro avg	0.97	0.96	0.96	334
weighted avg	0.96	0.96	0.96	334

Fig 2 : Classification report

In SVM, the support vectors are the data points that lie closest to the decision boundary or hyperplane.
Support vectors are the data points that have non-zero coefficients (alphas) in the solution to the optimization problem formulated by the SVM algorithm. During the training phase, the SVM algorithm identifies the support vectors as those instances that contribute to the determination of the optimal hyperplane. The instances that do not fall within the margin or on the wrong side of the hyperplane are typically the support vectors. The number of support vectors is an essential factor in understanding the complexity of the SVM model. A higher number of support vectors may indicate a more complex decision boundary, potentially leading to overfitting.
During the classification phase, when a new document is presented to the trained SVM model, it is classified based on its position relative to the decision boundary, influenced by the support vectors. Support vectors essentially act as representatives of their respective classes and contribute to the model's ability to generalize to unseen data.

	url	category_code	predict code
0	https://techcrunch.com/2022/02/18/coatue-backs...	4	0
1	https://www.bbc.com/sport/formula1/60431001	3	3
2	https://www.bbc.com/sport/cricket	3	3
3	https://indianexpress.com/section/india/politi...	2	2
4	https://indianexpress.com/section/entertainment/	0	1
5	https://indianexpress.com/section/business/	1	0
6	https://drive.google.com/file/d/1qMfkNYl-ybYQx...	3	0
7	https://drive.google.com/file/d/1YmFQ4HiUeJQjV...	3	0
8	https://www.bbc.com/news/technology/	4	4

Fig 3 : Categorized text

Predict code refers to the numeric labels assigned by the SVM classifier to represent the predicted categories of the input documents. Each category is mapped to a specific numeric code, facilitating model interpretation and evaluation.
The predicted category codes are compared to the actual category codes to evaluate the model's performance. Common evaluation metrics, such as accuracy, precision, recall, and F1-score, are calculated using the predicted and actual labels to assess the model's effectiveness in classifying documents. When testing the model on real-world data extracted from specific URLs, the "predict code" represents the model's predictions for the categories of the content found on those websites.

7. CONCLUSION

This paper introduces support vector machines for text categorization. It provides both theoretical and empirical evidence that SVMs are very well suited for text categorization. The theoretical analysis concludes that SVMs acknowledge the particular properties of text: (a) high dimensional feature spaces, (b) few irrelevant features (dense concept vector), and (c) sparse instance vectors.The experimental results show that SVMs consistently achieve good performance on text categorization tasks, outperforming existing methods substantially and
significantly. With their ability to generalize well in high dimensionalfeature spaces, SVMs eliminate the need for feature selection, making the application of text categorization considerably easier. Another advantage of SVMs over the conventional methods is their robustness. SVMs show good performance in all experiments, avoiding catastrophic failure, as observed with the conventional methods on some tasks. Furthermore, SVMs do not require any parameter tuning, since they can find good parameter settings automatically. All this makes SVMs a very promising and easy-to-use method for learning text
classifiers from examples.

8. FUTURE WORK

> Experiment with different SVM kernel functions beyond linear kernels, such as polynomial, radial basis function (RBF), or sigmoid kernels, to explore their impact on classification accuracy in text data.

>Investigate hybrid models combining deep learning architectures like Convolutional Neural Networks (CNNs) or Recurrent Neural Networks (RNNs) with SVMs to harness the strengths of both approaches for improved performance in text classification tasks.

>Develop strategies for incremental learning with SVMs to handle streaming data or large-scale datasets efficiently, improving scalability and adaptability of the model to evolving text data.

>Extend text classification to specific domains (e.g., medical, legal, or scientific texts) by tailoring feature representations and model architectures, addressing domain-specific challenges to achieve higher accuracy.

9. REFERENCES

[1] Karuna P. Ukey,Dr. A.S. Alvi "Text clssification using support vector machine" Department of I.T. PRMIT & R, Bandera Amravati, India Vol. 1 Issue 3, May - 2012 ,ISSN: 2278-0181 IEEE.

[2] S. Chowdhury and M. P. Schoen, "Research Paper Classification using Supervised Machine Learning Techniques," 2010 Intermountain Engineering, Technology and Computing (IETC).

[3] Crammer, K., & Singer, Y. (2003). Ultraconservative online algorithms for multiclass problems. This paper discusses SVM algorithms for multiclass classification, which is an important aspect of text classification when dealing with multiple categories.

[4] Joachims, T. (2002). Learning to Classify Text Using Support Vector Machines: Methods, Theory and Algorithms. While not a paper, this is a Ph.D. thesis by Joachims that provides a comprehensive overview of text classification using SVMs. It might contain valuable insights and references.