Performance Analysis of a Micromodel-based Multinomial Classifier

Jawahar Y V
B.Tech (IT)
Vellore Institute of Technology
Vellore, India
jawahar.yv2020@vitstudent.ac.in

Sanjay Nithin S
B.Tech (IT)
Vellore Institute of Technology
Vellore, India
sanjaynithin.s2020@vitstudent.ac.in

Dr. Varalakshmi M
School of Computer Science & Engg
Vellore Institute of Technology
Vellore, India
mvaralakshmi@vit.ac.in

Peer Mohideen
CapitalOne
San Jose, CA, USA
pupeer@gmail.com

Abstract— **Supervised learning methods demand large scores of labelled training data to achieve high predictive accuracy for classification problems. In contrast, the less accurate unsupervised learning models can work on raw data but fail to provide any useful insight. Micromodel-based multinomial classifier is a one-stop solution for increased performance with less data and less time. This paper investigates the effectiveness of micromodels for multiclass classification as against the existing supervised and unsupervised machine learning models and artificial neural networks. The model built using One-class Neural Network (OC-NN) to classify students' resumes into four different streams for admission into postgraduate programmes in top-ranked universities, outperforms some of the most powerful machine learning models.**

Keywords—micromodel, resume classification, multiclass classification, one-class neural network

I. INTRODUCTION

Multi-class classification is a frequently encountered task in supervised learning which classifies the given samples into one of several categories. Classification of tweets, movie reviews, product reviews, resumes and images are to quote a few of the applications that require multi-class classification. Logistic regression and Support Vector Machines are natural choices for binary classification. However, existing literature demonstrates the use of those algorithms' variants for multinomial classification. Logistic regression when used for classifying tweets into four broad topic categories has proved to increase the prediction accuracy [1] and when used along with Parts-of-Speech tagging helps for efficient classification of Amazon customers' reviews [2]. In a similar way, SVM has been used in disease prediction to predict diabetes and pre-diabetes conditions [3] and in image classification with low generalization error [4]. In the context of text classification, SVM used in conjunction with word2vec and kNN shows promising results [5][6].

Considering the probabilistic models, Bayesian NB classifier with Gaussian event model has outperformed the classical NB classifiers when applied for 20 newsgroups and WebKB data for text classification [7]. Sang-Bum Kim et al. [8] have applied per-document text normalization and feature weighting method for parameter estimation that greatly improves the performance of NB text classifier. In a Random Forest based classifier, Md Zahidul Islam et al. [9] have proposed generating diverse decision trees with each decision tree evaluated based on its prediction reliability that is a measure of the features used by the tree to make a prediction.

A comparative study of the various classifiers such as Naïve Bayes, Random Forest, Decision tree, SVM and Logistic Regression for text classification has been made out of which the logistic regression model achieves the highest accuracy [10]. In a related comparative analysis, Linear SVM has outperformed Logistic Regression and RF for resume classification and further a recommendation model based on k-Nearest Neighbours and Cosine Similarity aids in finding the CVs that match closely to the given job description [11]. Short text classification using k-NN results in better accuracy than NB and SVM [12].

Aytug Onan et al. have performed scientific text classification [13] with five different ensemble techniques namely Adaboost, Dagging, Bagging, Majority Voting and Random Subspace using NB, SVM, Random Forest and Logistic Regression as the base models. They have also evaluated the effect of different statistical keyword extraction methods on these classification algorithms. Their results show that the most-frequent-measure based extraction method applied for a Random Forest, a Bagging ensemble, yields the most optimal results. In yet another study related to resume classification, a voting ensemble of Logistic Regression, Multinomial and Bernoulli NB and Linear SVC has shown better prediction results compared to the individual base models [14].

Attempts made to use NLP techniques and regular expressions to match sections of resumes with appropriate sections of job posts has contributed to performance improvement in resume classification [15]. Similarly, Random Forest model results in better predictions when the feature set is obtained using TF-IDF and Text Rank algorithms [16].

Recent studies illustrate the use of deep neural networks such as RNN, CNN and LSTM for short-text classification [17]. Evolutionary Contiguous Convolutional Neural Network based on Differential Evolution and Particle-Swarm Optimization based deep neural network have demonstrated high classification accuracy [18]. This is attributed to the ability of these models to use information from the preceding texts.

The success of any deep neural network for multi-class classification depends on the labelled training dataset. The dataset should be large, balanced and representative of each class, lest the model will fail to identify a specific class. More the number of classes that a model should handle, more should be the training data. It may not be possible to

979-8-3503-2028-2/22/$31.00 ©2022 IEEE
DOI 10.1109/CSCI58124.2022.00152

generate enough and appropriate synthetic training data for all applications. Thus, collecting huge amount of labelled and balanced dataset is one of the major challenges in supervised learning. The next challenge is when a new class is tried to be added to an already trained network. It should be ensured that the samples for the new class is balanced and the model should be retrained with the whole set of training data including the new samples. These limitations can be addressed using micromodels.

Micromodels are smaller and compact models each of which can identify samples belonging to a particular class. They are capable of delivering more precise analytics than macro-models but with much lesser time and data. In certain applications, building several micromodels is easier and more efficient than building a single, large model (macro model), as they minimize the efforts to collect, clean and annotate large volumes of data.

In general, in a micromodel based multi-class classifier system, the number of micromodels used should be equal to the number of classes to be detected and distinguished. Each micromodel is a single class classifier and so, each can be individually and parallelly trained with samples of that class. After training, the test instances can be fed to these micromodels, the results of which are fed to the comparator / aggregator module to identify that micromodel which resulted in the highest probability value and the corresponding class which the test instance belongs to. The advantage of micromodel-based architecture is the ease with which the model can be extended to handle new classes in future. The classifier system can be easily edited to handle these modifications by simply adding or removing a micromodel to /from the existing set of micromodels. It is adequate to train only the newly added micromodels; in the operational phase, the instances are routed to the entire set of micromodels including the newly added ones and the comparator module should consider including the additional probability values returned by the newly added micromodels, before producing the results [19][20].

These micromodels are different from ensemble models which combines several (weak) base models to make a single prediction by averaging the results of the individual learners. Micromodels on the other hand, make their own individual predictions and are not weak learners. Research works on micromodels is very limited and still in the infancy level. In an attempt to investigate the effectiveness of micromodels for multiclass classification as against the existing supervised and unsupervised machine learning models and artificial neural networks, this paper focuses on implementing these algorithms and studying their comparative performance for multi-class classification of student resumes for higher studies.

In the recent years, there is a steep rise in the number of students applying for higher studies in top-ranked universities from all corners of the world, with the intent to hone their skills and make successful careers. Every year, numerous applications are received for the various programmes offered in these universities. It involves painstaking efforts to analyse the resumes and select the matching one for a specific programme. Inspired by the promising results of the micromodels in a few of the domains, we have developed a micromodel-based classifier to segregate the students' resumes into different streams based on the skills and competencies acquired in their undergraduate studies. The results are compared with those of the other popular machine learning and deep learning methods. This classifier tool can help the higher education web portals to serve better and the university admissions committee to expedite the selection process in a more efficient and customized way. To the best of our knowledge, resume screening for higher studies is not explored, although extensive research has been done in classifying the job resumes to minimize the efforts of recruiters and job portals.

The rest of the paper is organized as follows. Section 2 discusses the implementation of the proposed micromodel architecture for the chosen application. Section 3 presents the comparative results of other machine learning and deep learning techniques. Section 4 outlines the conclusions from our analysis and proposes ideas for future work.

II. MICROMODEL-BASED MULTI-CLASS CLASSIFIER

A. Dataset Preparation

With an aim to work with real datasets, resumes of final year undergraduate engineering students are collected. Owing to the unstructured nature of the data in the resumes, affinda resume parser is used to analyse them and generate a JSON file that contains the data structured as key-value pairs. The skills enumerated in each JSON file are collected to build a list of 1000 and more unique skills which constitutes our feature set. Using this feature set, one-hot encodings of the individual samples are generated to form the actual input dataset. The four most common postgraduate programmes in computer science offered in the top-ranked universities that match the students' skillsets are selected as the four classes in our application – MS in Computer Science, MS in Cloud Computing, MS in Data Science and MS in Cyber Security. The resumes are carefully scrutinized and annotated manually with one of these four classes.

Dataset is balanced using SMOTE and dimensionality reduction is performed with PCA after which 20% of the total samples are reserved for testing and the remaining 80% of the samples are split into four groups with each group consisting of samples belonging to one particular class.

B. Model Architecture

Four individual micromodels are built using one-class neural network (OC-NN), a network proposed for anomaly detection [21], to have one-to-one correspondence between the models and the output classes. Fig.1 shows the architecture of a OC-NN based micromodel that includes a deep autoencoder, trained using the input and this pre-trained encoder is fed as input to the feed-forward network with a single hidden layer and linear activation function. Each micromodel is trained with samples belonging to one specific stream. The comparator module collects the probability values returned by the individual micromodels and generates the output as the class corresponding to the micromodel that produced the highest probability value.

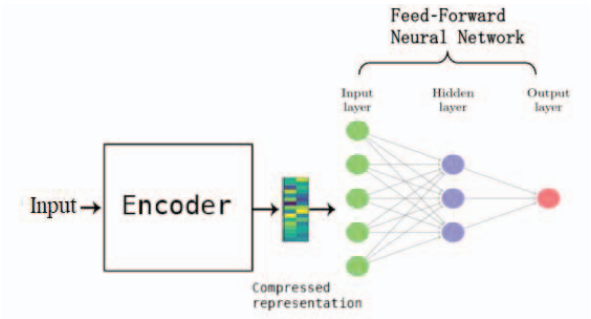

Fig. 1. One-Class Neural Network based Micromodel

III. RESULTS AND DISCUSSIONS

The proposed model gives an accuracy of 87.6%. Confusion matrix for the proposed model is shown in Fig. 2.

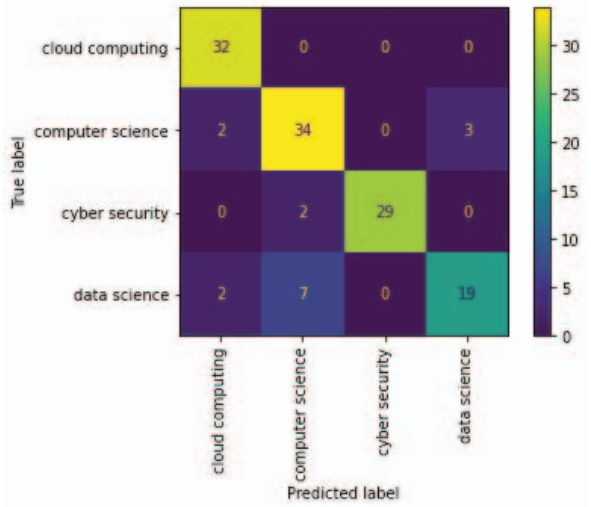

Fig. 2. Confusion Matrix for Micromodel-based Multinomial Classifier

In order to study the reliability of unsupervised learning for resume classification to provide future directions to use the abundantly available, unlabelled data to make predictions, an enhanced *k*-mode clustering algorithm is implemented, by adopting a similar approach followed in *k*-means++ algorithm. Instead of choosing the initial *k* cluster centroids randomly, it chooses the first centroid randomly and based on the distance of the other points to the nearest, previously chosen centroid, the other centroids are selected. After identifying the initial clusters, the implementation proceeds with the standard k-mode clustering algorithm. The performance of the enhanced *k*-mode clustering algorithm is measured by labelling all the points in a particular cluster with the majority class of that cluster and comparing them with the truth values. Fig. 3 presents the confusion matrix for the enhanced *k*-mode algorithm.

Even with a careful choice of *k* value, the model is found to be 44.75% accurate. Results clearly indicate that even in domains with scarce annotated-data, unsupervised learning is not a very viable solution.

Performance of the proposed micro-model based multiclass classifier is evaluated by comparing its results with the most popular supervised machine learning models. SVM, Logistic Regression, Gaussian Naïve Bayes and Random Forest are the models considered for assessment. After hyperparameter tuning, the models achieved an accuracy of 85%, 82%, 61% and 83% respectively. The dependency among a few of the input features explains the low accuracy level for NB. In addition to the machine learning models, a fully-connected feed forward neural network is also implemented to compare and assess the performance of the proposed model in a similar test environment. The network is built with 3 hidden layers and with 'softmax' activation function in the output layer and 'softplus' activation function in the hidden layers. Its performance is close to SVM with an accuracy of 85%. Fig.4 depicts the various performance metrics such as accuracy, recall, precision and F1 score for the different models. The proposed model outperforms all the other supervised learning models including the traditional feed forward neural network. Although there is no significant difference between SVM and the proposed system, still it is better than SVM.

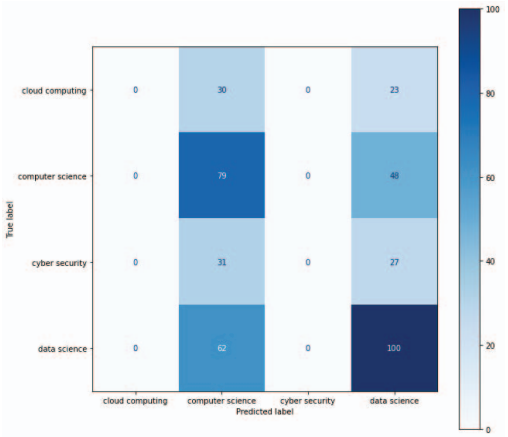

Fig. 3. Confusion Matrix for Enhance *k*-mode Cluster

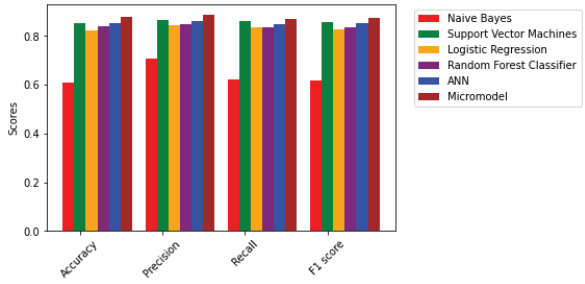

Fig. 4. Performance Metrics Scores for the different Models

IV. CONCLUSION

Supervised learning models are successful only when trained with large amount of annotated data. Furthermore, for multiclass classification problems, data should be balanced and representative of each class. In low-resource domains where it is impossible to get large volumes of labelled data, unsupervised techniques can be used to discover hidden patterns but at the cost of performance. Micromodel-based multiclass classifier is a one-stop solution for increased performance with less data and less time. The model built to classify students' resumes into four

different streams for admission into postgraduate programmes in top-ranked universities outperforms some of the most powerful machine learning models. Applicability of micromodels in other domains can be explored and the idea can be extended to multilabel classification.

REFERENCES

[1] Indra, S. T., Liza Wikarsa, and Rinaldo Turang. "Using logistic regression method to classify tweets into the selected topics." 2016 international conference on advanced computer science and information systems (icacsis), pp. 385-390, IEEE, 2016.

[2] Pranckevičius, Tomas, and Virginijus Marcinkevičius. "Application of logistic regression with part-of-the-speech tagging for multi-class text classification." 2016 IEEE 4th workshop on advances in information, electronic and electrical engineering (AIEEE), pp. 1-5, IEEE, 2016.

[3] Yu, Wei, et al. "Application of support vector machine modeling for prediction of common diseases: the case of diabetes and pre-diabetes." BMC medical informatics and decision making 10.1 (2010), pp.1-7.

[4] Chapelle, Olivier, Patrick Haffner, and Vladimir N. Vapnik. "Support vector machines for histogram-based image classification." IEEE transactions on Neural Networks 10.5 (1999), pp. 1055-1064.

[5] Lilleberg, Joseph, Yun Zhu, and Yanqing Zhang. "Support vector machines and word2vec for text classification with semantic features." 2015 IEEE 14th International Conference on Cognitive Informatics & Cognitive Computing (ICCI* CC), pp. 136-140, IEEE, 2015.

[6] Lin, Yun, and Jie Wang. "Research on text classification based on SVM-KNN." 2014 IEEE 5th International Conference on Software Engineering and Service Science, pp. 842-844, IEEE, 2014.

[7] Xu, Shuo. "Bayesian Naïve Bayes classifiers to text classification." Journal of Information Science 44.1 (2018), pp. 48-59.

[8] Kim, Sang-Bum, et al. "Some effective techniques for naive bayes text classification." IEEE transactions on knowledge and data engineering 18.11 (2006), pp. 1457-1466.

[9] Islam, Md Zahidul, et al. "A semantics aware random forest for text classification." Proceedings of the 28th ACM international conference on information and knowledge management, pp. 1061-1070, 2019.

[10] Pranckevičius, Tomas, and Virginijus Marcinkevičius. "Comparison of naive bayes, random forest, decision tree, support vector machines, and logistic regression classifiers for text reviews classification." Baltic Journal of Modern Computing 5.2 (2017), pp. 221.

[11] Roy, Pradeep Kumar, Sarabjeet Singh Chowdhary, and Rocky Bhatia. "A Machine Learning approach for automation of Resume Recommendation system." Procedia Computer Science 167 (2020), pp. 2318-2327.

[12] Khamar, Khushbu. "Short text classification using kNN based on distance function." International Journal of advanced research in computer and communication engineering 2.4 (2013), pp. 1916-1919.

[13] Onan, Aytuğ, Serdar Korukoğlu, and Hasan Bulut. "Ensemble of keyword extraction methods and classifiers in text classification." Expert Systems with Applications 57 (2016), pp. 232-247.

[14] Gopalakrishna, Suhas Tangadle, and Vijayaraghavan Vijayaraghavan. "Automated tool for Resume classification using Sementic analysis." International Journal of Artificial Intelligence and Applications (IJAIA) 10.1 (2019).

[15] Zaroor, Abeer, Mohammed Maree, and Muath Sabha. "A hybrid approach to conceptual classification and ranking of resumes and their corresponding job posts." International Conference on Intelligent Decision Technologies, Part I 9, pp. 107-119, Springer, Cham, 2017.

[16] Sun, Yanxiong, et al. "Application research of text classification based on random forest algorithm." 2020 3rd International Conference on Advanced Electronic Materials, Computers and Software Engineering (AEMCSE), pp. 370-374, IEEE, 2020.

[17] Lee, Ji Young, and Franck Dernoncourt. "Sequential short-text classification with recurrent and convolutional neural networks." arXiv preprint arXiv:1603.03827 (2016).

[18] Prabhakar, Sunil Kumar, et al. "A Framework for Text Classification Using Evolutionary Contiguous Convolutional Neural Network and Swarm Based Deep Neural Network." Frontiers in Computational Neuroscience (2022).

[19] Speiser, Leonard Robert, and Joel T. Kaardal. "Computer vision classifier using item micromodels." U.S. Patent No. 11,126,898. 21 Sep. 2021.

[20] Lee, Andrew, et al. "Micromodels for efficient, explainable, and reusable systems: A case study on mental health." arXiv preprint arXiv:2109.13770 (2021).

[21] Chalapathy, Raghavendra, Aditya Krishna Menon, and Sanjay Chawla. "Anomaly detection using one-class neural networks." arXiv preprint arXiv:1802.06360 (2018).