IJCRT.ORG ISSN : 2320-2882

INTERNATIONAL JOURNAL OF CREATIVE RESEARCH THOUGHTS (IJCRT)

An International Open Access, Peer-reviewed, Refereed Journal

A Machine learning approach for Industry classification Using resume data

Sukanya V N

Justin Seby

Abstract: **Recruitment in the Information Technology and Computer Science sector has seen an exponential increase in recent times.** ***Since there is an abundance of resumes in different styles and formats from job seekers, it isn't easy to screen. In this context, we put forward an approach to classifying and categorizing resumes using different machine learning algorithms, SVM classifier, Naïve Bayes, and Logistic Regression. The resumes are in PDF format. From these, the resumes are classified into twenty-five different industries. For this, the unstructured resumes are converted to structured resumes using different NLP task. The extracted features are vectorized using techniques such as TF-IDF. The vectorized features are given to different classifiers. The classifier categorizes the candidate resumes into the corresponding job categories. Bernoulli Naïve Bayes ,The accuracy of the four models are calculated and selecting the most accurate model to categorize.***

I. INTRODUCTION

Nowadays, job recruitments through the internet are more prominent and beneficial to both employers and employees. Because the applicants could make a formal application without being present for the interview, it saves both recruiters and the applicant's time. However, many resume templates are available on the internet nowadays. To sort these resume will be a cumbrous task. Most of the applicants try to gain the attention of the recruiters by using different formats to build the resumes. And collect the information from other resume formats and selecting the candidate are getting troublesome. To get rid of this problem, we put forward our project, the Resume Classifier. A comparative study is done on four classification models, such as Bernoulli NB, Naive Bayes, Logistic Regression, and SVM (Support Vector Machine), to find out the best among them. We aim to find the resume classifier that would categorize any resume set into 25 predefined different categories with the most accurate results. Firstly, all the resumes are converted into text format. Since the attendees do not follow a standard format, the unstructured resume files are transformed into a structured standard format using NLP tasks such as Bag of Words and Pos tagging. And the resumes are Tokenized (the process of splitting a large sample of text into words) and Vectorized (this process provide unique vector values to each word). The Natural Language Toolkit (NLTK) is a library used to achieve this.

II. RELATED WORKS

Recently the machine learning techniques have been broadly used in diverse fields. In the competing era of technology, applying these machine learning techniques to the category based job post classifications is focused on many pieces of research. A resume classifier is a text-based classification system Sam Scott and Stan Matwin presents several alternative ways to represent text based on syntactic and semantic relationships between words. The feature generation technique is described in several machine learning models such as Support Vector Machine (SVM) and Gradient Boosted Decision Trees (GBDT). The authors in point out that the popular methods for applying SVMs to multiclass classification problems usually decompose the multiclass problems into several two-class issues that can be addressed directly using several SVMs. Machine learning tasks of the text classification process include three steps: presentation of text documents, preparing the classifier for text documents, and evaluation of the classifier demonstrated in. A machine learning-based document classification approach requires labelling documents with predefined classes to create a set of training data. The IEEE conference paper [6] provides that this training data is then used to learn a model that can assign one or more of the predefined classes to new documents. Various researches are conducted to discuss different classification strategies related to the job post-classification problem. Recent studies describe SVM, Random Forests (RF) and Extreme Gradient Boosting(XGB) and Neural Networks for job matching. The four classification algorithms that used here in this project is discussed briefly.

A. Naive Bayes

A Naive Bayesian model is easy to build, with no complicated iterative parameter estimation which makes it particularly useful for very large datasets. The Naive Bayesian classifier is based on Bayes' theorem with the independence assumptions between predictors. Bayes theorem provides a way of calculating the posterior probability, P(c|x), from P(c), P(x), and P(x|c). Naive Bayes classifier works based on class conditional independence assumption, which assumes that the effect of the value of a predictor (x) on a given class (c) is independent of the values of other predictors[5].

-P(c) is the prior probability of class.
-P(x|c) is the likelihood, which is the probability of predictor given class.
-P(x) is the prior probability of predictor.

The distribution is parametrized by vectors $\theta y=(\theta y1,\ldots,\theta yn)$for each class y, where n is the number of features and θyi is the probability P(xi|y) of feature i appearing in a sample belonging to class y. The parameter θy is estimated by a smoothed version of maximum likelihood, i.e. relative frequency counting:

$$\theta\hat{}yi = Nyi + \alpha Ny + \alpha n$$

Where $Nyi = \sum x \in T xi$ is the number of times feature i appears in a sample of class y in the training set T, and $Ny\sum i=1nNyi$ is the total count of all features for class y. The smoothing priors $\alpha \geq 0$ account for features not present in the learning samples and prevent zero probabilities in further computations. Setting $\alpha=1$ is called Laplace smoothing, while $\alpha<1$ is called Lidstone smoothing. Naive Bayes for the classification of resumes into their corresponding job categories is implemented with the module sklearn.naive_bayes.MultinomialNB from the sklearn package where the data are typically represented as word vector counts or TF-IDF vectors [6] .

B. Logistic Regression

A model that is used to predict the probabilities of the different possible outcomes of a categorically distributed dependent variable, given a set of independent variables (which may be real-valued, binary-valued, categorical-valued, etc.).Logistic regression for resume classifier is implemented with sklearn.linear_model.LogisticRegression module in the sklearn package, which is a multiclass case and the training algorithm uses the one-vs-rest (OvR) scheme, separate binary classifiers are trained for all classes. This class implements regularized logistic regression using the 'liblinear' library, 'newton-cg', 'sag' and 'lbfgs' solvers. It can handle both dense and sparse input.

Here we are using scikit logistic regression of multinomial logistic regression with optional L2 regularization. binary class L2 penalized logistic regression minimizes the following cost function:

C. Support Vector Machine

A support vector machine algorithm which can be used for classification constructs a set of hyperplanes in a high dimensional space in which each dimension specifies the features that can be used for classification.[3] Data points are viewed as $(\vec{x},y)$ tuples, $\vec{x}=(x1,\ldots,xp)$ where the xj is the feature values and y is the classification (usually given as +1 or −1). Optimal classification occurs when such hyperplanes provide maximal distance to the nearest training data points. Intuitively, this makes sense, as if the points are well separated, the classification between the two groups is much clearer [4].
Linear SVC, one of the SVM classifier capable of performing multi-classification through one-vs-the-rest strategy which involves training a single classifier per class, with the samples of that class as positive samples and all other samples as negatives.

SVM classifier is implemented with sklearn.svm.linearsvc module which is imported from the sklearn package and is used for the classification of resumes into their corresponding one out of twenty-five categories. This class supports both dense and sparse input

D. *Bernoulli Naïve Bayes*

Bernoulli Naive Bayes is a type of Naive Bayes. A Bernoulli NB model was chosen due to the relevance of using binary features in a two-class classification problem [2]

III. METHODOLOGY

The main experiment proposed in the work is job prediction. Here we select about 2550 sets of resumes which belongs to 25 different set of categories. From the total 70% of resume are taken as training data and remaining for test data. We considered four training models, Bernoulli NB, Support Vector Machine(SVM), Naive Bayes and Logistic Regression. Since the four models support numeric and binary features, all the categorical features are converted into binary features. So first, we did data cleaning processing by using the python package pandas. The dataset is in the csv file form. The dataset contains four fields ID, category and resumes. In the data cleaning process, if the category field or resume field is null, it will be neglected. After that, the preprocessing of the dataset was done. Preprocessing consists of tokenization and vectorization. Tokenization and vectorization were done using the nltk tool kit, which is provided by the python. Tokenization converts the data into tokens. Vectorization converts each token into vectors (each token have a corresponding vector in vector space). After that, the vectors are given into the four models.

After the training of the four model; Bernoulli NB, Support Vector Machine (SVM), Logistic Regression and Naive Bayes, Bernoulli NB shows more accuracy among the four models followed by SVM. So we continued our further process of model training by Bernoulli NB, because other models found to give less accuracy. By a detailed study of the process, it found that data distribution was not normalized. Some data are occurring in huge amount, while some are in less amount (eg, the category engineering has a huge amount of data while the apparel category has less amount of data).

So the solution to the problem was either to replicate the dataset or to integrate the dataset. The integration of the dataset created complications. So we chose to replicate the dataset, and the dataset was replicated to a standard number. Before replication, there are 1544 sets of resumes. After replication, it turned around 2400 sets of resumes. The main problem we thought of replication was overfitting. But after the replication, our Bernoulli NB model gives about 85% of accuracy without overfitting of data. The table showed that among the four models Bernoulli NB, SVM, Naïve Bayes and Logistic Regression, Bernoulli NB shows the most accuracy for the same training and testing data. So from the comparative study of these four models, Support Vector Machine is chosen for classification of resumes into their corresponding twenty-five predefined categories. These results are displayed below

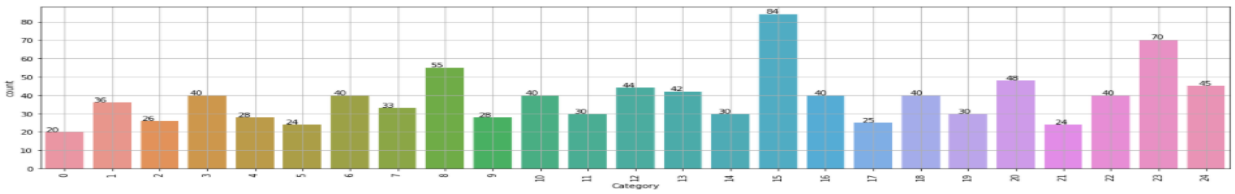

Figure 1: The number resumes of each industry

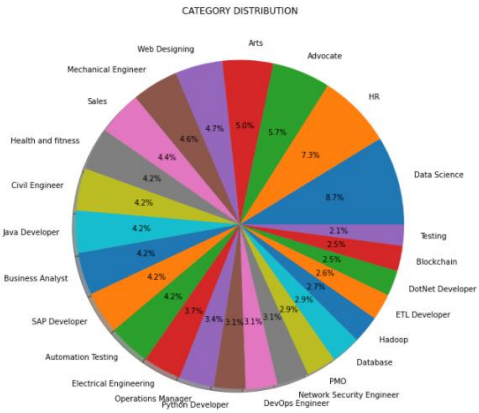

Figure 2: Category classification with normalization

IV. RESULTS AND DISCUSSION

The results from the model are promising because the resume classifier application successfully automates the manual task of project allocation to the new recruits of the organization based on the field and work experience mentioned by the candidate in the profile. The table shows the accuracy given by each model that was used in this project. The Bernoulli NB classifier performs exceptionally well compared to the other classifier models we were using. Bernoulli NB gives 85% accuracy compared to the other models, which offers about 65-70% accuracy. In the future, we like to build an ensemble deep learning model [1], which gives even more precision and accuracy.

Models	Accuracy
Bernoulli NB	85%
Support Vector Machine(SVM)	80%
Logistic Regression	75%
Naive Bayes	65%

Table 4.1: Comparisons between the four models

REFERENCES

[1] Z.-H. Zhou, J. Wu, W. Tang, "Ensembling neural networks: Many could be better than all", Artificial Intelligence, vol. 137, no. 1-2, pp. 239-263, 2002.**[2]** L. Sayfullina et al., "Efficient Detection of Zero-day Android Malware Using Normalized Bernoulli Naive Bayes," 2015 IEEE Trustcom/BigDataSE/ISPA, 2015, pp. 198-205, doi: 10.1109/Trustcom.2015.375.

[3] I. Guyon, B. Boser, V. Vapnik, "Automatic Capacity Tuning of Very Large VC-dimension Classifiers", 1993

[4] H. Zhang, "The Optimality of Naive Bayes",Proc.Flaris

[5] Rennie, J . D, Shih, Teevan and Karger D, "Tracking the poor assumptions of naive bayes text classifiers", In ICML (vol.3) .

[6] V. Metsis, I.Androutsopoulos and G. Paliouras, "Spam filtering with Naive Bayes?", 3rd conf. on Email and Anti-spam(CEAS).

[7] Faizan Javed, Qinlong Luo, Matt McNair, Ferosh Jacob, Meng Zhao, Tae Seung Kang, " Carotene: A Job Title Classification System for the Online Recruitment Domain".