© March 2018 | IJIRT | Volume 4 Issue 10 | ISSN: 2349-6002

# Prediction of Industry Readiness among Job Seekers Using Bagging Algorithm

S.Prathibha Sai[1], Mrs.S.Sajida[2]
[1]Student, Dept. of MCA, KMM Institute of Post Graduate Studies
[2]Assistant Professor, Dept. of MCA, KMM Institute of Post Graduate Studies, Tirupati, A.P

***Abstract-*** **The Industry Readiness Program is a short - term intensive non-credit training program developed with industry employers to prepare students for entry to mid- level jobs in the local, diverse advanced manufacturing industry. In this paper, we mainly aim to find what are the techniques we should be considered for better performance of the employee in industries and what are the different factors that affect employee performance in an organization. Job readiness is training a participant receives to prepare them to seek or obtain employment, and to keep their jobs once they are hired. We will predict that what are the different jobs that an employee should be well trained in understanding the training needs. We have different algorithms for predict for an industry readiness among job seekers like NaviBayes, Neural networks, bagging algorithms is available. Among these, we are using the Bagging Algorithm for predicting the results. Because the Bagging Algorithm will give better accuracy and performance than compared with the other algorithms.**

***Index Terms-*** **NaviBayes, Neural networks, bagging algorithms, job seekers.**

## I. INTRODUCTION

Job Search and Job Readiness Assistance mean the act of seeking or obtaining employment and preparation to seek or obtain employment. For federal participation purposes job search and job readiness assistance is a single component. This activity must be supervised no less frequently than daily. Job Readiness must be a structured and supervised program and includes two types of activities. Preparation for seeking or obtaining employment. This includes activities such as preparing a resume or job application, training in interviewing skills, instruction in work place expectations, training in effective job seeking, and life skills training. Substance abuse treatment, mental health treatment, or rehabilitation activities for those who are otherwise employable. Job Search must be a structured and supervised activity which may include the following: Making contacts with employers by phone, making contacts in person, Use of the Internet to learn of suitable job openings, applying for jobs, and interviewing for jobs.

## II. RELETED WORK

Bagging predictors is a method for generating multiple versions of a predictor and using these to get an aggregated predictor. The aggregation averages over the versions when predicting a numerical outcome and does a plurality vote when predicting a class. The muntiple versions are formed by making bootstrap replicates of the learning set and using these as new learning sets. Tests on real and simulated data regression trees and subset selection in linear regression show that bagging can give sub stantial gains in accuracy. The vital element is the instability of the prediction metho. If perturbing the learning set can cause significant changes in the predictor constructed, then bagging can improve accuracy(20).

To improve the performance of weak regression and classification rules, a number of combining techniques can be used. During the last few years, the most popular methods have become bagging, boosting and the random subspace method. They all modify the training data set, build classifi- ers on these modified training sets, and then combine them into a final decision rule by simple or weighted majority voting. However, they perform in a different way(21).

## III. BAGGING ALGORITHM

Bagging is a method for improving results of machine learning classification algorithms. This method was formulated by Leo Breiman and its name was deduced from the phrase "bootstrap aggregating" In case of classification into two possible classes, a classification algorithm creates a classifier H: D Æ {-1, 1} on the base of a training set of example descriptions (in our case played by a document collection) D. The bagging method creates a sequence of classifiers Hm, m=1… M in respect to modifications of the training set. These classifiers are combined into a compound classifier. The prediction of the compound classifier is given as a weighted combination of individual classifier predictions:

$$H(d_i) = sign\left(\sum_{m=1}^{M} \alpha_m H_m(d_i)\right).$$

The meaning of the above given formula can be interpreted as a voting procedure. An example di is classified to the class for which the majority of particular classifiers vote. Articles [2] and [6] describe the theory of classifier voting. Parameters αm, m=1,…,M are determined in such way that more precise classifiers have stronger influence on the final prediction than less precise classifiers. The precision of base classifiers Hm can be only a little bit higher than the precision of a random classification. That is why these classifiers Hm are called weak classifiers.

We experimented with the following bagging algorithm:

A bagging algorithm for multiple classifications into several classes.

1. Initialization of the training set D
2. for m = 1... M.

2.1. Creation of a new set Dm of the same size |D| by random selection of training examples from the set D (some of examples can be selected repeatedly and some may not be selected at all).

2.2. Learning of a particular classifier Hm: Dm → R by a given machine learning algorithm based on the actual training set Dm.

3. Compound classifier H is created as the aggregation of particular classifiers Hm: m = 1, ...,M and an example Di is classified to the class cj in accordance with the number of votes obtained from particular classifiers Hm

$$H(d_i, c_j) = sign\left(\sum_{m=1}^{M} \alpha_m H_m(d_i, c_j)\right)$$

If it is possible to influence the learning procedure performed by the classifier Hm directly, classification error can be minimized also by Hm while keeping parameters αm constant.

The above described algorithm represents an approach called base version of bagging. There are some other strategies called bagging like strategies which work with smaller size of the training set of example descriptions. These strategies use a combination of the bagging method and the cross-validation method. The cross-validation represents the division of the training set into N subsets of D/N size. One of these subsets is used as the training set and the other subsets play the role of test sets.

In "bagging like strategies" the original training set is divided into N subsets of the same size. Each subset is used to create one classifier – a particular classifier is learned using this subset. A compound classifier is created as the aggregation of particular classifiers. The most known methods are: disjoint partitions, small bags, no replication small bags and disjoint bags. An illustrative example of the subset selection process to form new training subsets from an original one is presented in the rest of this section. The original training set containing sixteen examples is depicted in Figure 1.

A B C D E F G H I J K L M N O P

Figure 1 : Original training set D

The method of disjoint partitions uses random selection to select examples. Each example is selected only once. An example of four new subsets, created from the original training set in Figure 1, is presented in Figure 2. In general, if N subsets are created from the original training set, then each of them contains 1/N part from the original set. Union of particular subsets equals the original training set. For very large original set, partitions enable parallel learning of base classifiers.

A B C D | E F G H | I J K L | M N O P

Figure 2 : Disjoint partitions

Classifier H obtained from the aggregation of particular classifiers Hm learnt on disjoint partitions, achieves the best results from all „bagging like strategies". In the method of small bags, each subset is generated independently from the other subsets by random selection of training examples with the possibility to select an example repeatedly. An example can be located in several subsets and/or several times in one subset as well. The training sets illustrated in Figure 3 were obtained from the original set in Figure 1. Union of particular partitions does not guarantee to provide the original training set. Classifiers using the small bags reach the worst results from all „bagging like strategies".

| ACHL | BPLP | DIOH | KCFK |
|---|---|---|---|

Figure 3 : Small bags

In the method of no replication small bags, each subset is generated independently from the other subsets by random selection of training examples without any replication of examples. An example can occur in one subset, several subsets, or no subset. If it occurs in a subset, then exactly one copy is included in the subset. The training sets illustrated in Figure 4 were obtained from the original set in Figure 1. Union of particular partitions does not guarantee to represent the original training set.

| ACHL | OPLN | DIOH | KCFP |
|---|---|---|---|

Figure 4 : No replication small bags

The last method from the above mentioned ones is the method of disjoint bags. In this method, size of each subset does not equal |D| but is (slightly) greater. First, examples which occur in the original training set are distributed into subsets. Selection of training examples is performed in the same way as in the method of "disjoint partitions". Then, one or more examples are randomly selected and replicated within each subset. The number of replications has to be the same in each subset. An example of resulting division of training examples is illustrated in Figure 5. Each example from the original set occurs (once or more times) exactly in one subset.

| ABCDC | EFGHE | IJKLJ | MNOPO |
|---|---|---|---|

Figure 5 : Disjoint bags

Union of particular partitions does not provide the original training set. Classifiers using "disjoint bags" are known to reach the same or sometimes better results as those classifiers using „disjoint partitions".

## IV. CONCLUSION

In this paper we are use bagging algorithm, by this algorithm we are improves performance and accuracy compared to other algorithms to predict for industry among job seekers. Job readiness is training a participant receives to prepare them to seek or obtain employment, and to keep their jobs once they are hired. And bagging algorithm give better results.

## REFERENCES


[1] Jain AK, Chandrasekaran B. Dimensionality and sample size considerations in pattern recognition practice. In: Krishnaiah PR, Kanal LN (Eds) Handbook of Statistics, vol 2. North Holland, Amsterdam, 1987; 835–855.

[2] Friedman JH. Regularized discriminant analysis. J Am Statistical Assoc 1989; 84: 165–175

[3] An G. The effects of adding noise during backpropagation training on a generalization performance. Neural Computation 1996; 8: 643–674

[4] Breiman L. Bagging predictors. Machine Learning J 1996; 24(2): 123–140

[5] Freund Y, Schapire RE. Experiments with a new boosting algorithm. Proceedings 13th International Conference on Machine Learning 1996; 148–156

[6] Ho TK. The Random subspace method for constructing decision forests. IEEE Trans Pattern Analysis and Machine Intelligence 1998; 20(8): 832–844

[7] Efron B, Tibshirani R. An Introduction to the Bootstrap. Chapman & Hall, New York, 1993

[8] Schapire RE, Freund Y, Bartlett P, Lee W. boosting the margin: a new explanation for the effectiveness of voting methods. Ann Statistics 1998; 26(5): 1651–1686

[9] Breiman L. Arcing classifiers. Ann Statistics 1998; 26(3): 801–849

[10] Dietterich TG. An experimental comparison of three methods for constructing ensembles of decision trees: bagging, boosting, and randomization. Machine Learning 2000; 40(2): 139–157 11. Breiman L. Random forests – random features. Technical Report 567, University of California, Berkley, 1999

[11] Avnimelech R, Intrator N. Boosted mixture of experts: an ensemble learning scheme. Neural Computation 1999; 11: 483–497

[12] Schwenk H, Bengio Y. Boosting neural networks. Neural Computation 2000; 12: 1869–1887

[13] Alkoot FM, Kittler J. Population bias control for bagging k-NN experts. Proceedings Sensor Fusion: Architectures, algorithms and applications V. Orlando, FL, April 2001

[14] Skurichina M, Duin RPW. Bagging for linear classifiers. Pattern Recognition 1998; 31(7): 909–930.

[15] Fukunaga K. Introduction to Statistical Pattern Recognition. Academic Press, 1990

[16] Raudys Sˇ, Pikelis V. On dimensionality, sample size, classification error and complexity of classification algorithm in pattern recognition. IEEE Trans Pattern Analysis and Machine Intelligence 1980; 2(3): 242–252

[17] Bauer E, Kohavi R. An empirical comparison of voting classification algorithms: bagging, boosting and variants. Machine Learning 1999; 36(1/2): 105–142

[18] Raudys Sˇ, Duin RPW. On expected classification error of the Fisher linear classifier with pseudo-inverse of the covariance matrix. Pattern Recognition Letters 1998; 19(5–6): 385–392

[19] Cortes C, Vapnik V. Support-vector networks. Machine Learning 1995; 20: 273–297

[20] https://www.semanticscholar.org/paper/Bagging-PredictorsBreiman/046b7f6b48e4d9fcf173dea0a0802d7e87b383e1

[21] http://rduin.nl/papers/paa_02_bagging.pdf