

Performance Analysis of Machine Learning Algorithms in Resume Recommendation Systems



Inspiring Excellence

SUBMISSION DATE: 25.03.18

SUBMITTED BY:

Ibteaz Hasan (14301029)

Ratnadeep Chakraborty (14301075)

Md. Ashraful Alam (14301001)

Department of Computer Science and Engineering

Supervisor:

Amitabha Chakrabarty, Ph.D

Assistant Professor

Department of Computer Science and Engineering

Declaration

We, hereby declare that this thesis is based on results we have found ourselves.

Materials of work from researchers conducted by others are mentioned in references.

Signature of Supervisor

Signature of Authors

Amitabha Chakrabarty, Ph.D

Assistant Professor

Department of Computer Science and

Engineering

BRAC University

Ibteaz Hasan

(14301029)

Ratnadeep Chakraborty

(14301075)

Md. Ashraful Alam

(14301001)

ABSTRACT

We present an evaluation of machine learning algorithms on a model prepared by us for improving the recruitment processes of organizations. The recruitment of candidates, being an important process for any organization, entails the hiring of employees that would be best fit for the job and ultimately beneficial for them. We have taken resumes of candidates of an organization and extracted the attributes (namely academics, qualifications, etc. to name a few) and assessed them according to a scale and a corresponding scoring system to train our system so that the candidates with the best scores can be shortlisted. We applied algorithms like decision tree, support vector machine, multi-linear regression and Bayesian ridge regression to train our system. Of all these the best results were given by decision tree and support vector machine regression.

Keyword: Resume, machine learning, recruitment, regression.

Acknowledgement

We would like to thank our supervisor Dr. Amitabha Chakrabarty, for his immense support and guidance we could finish our research properly. He was always there for us and we learned a lot from him that helped us to reach our goal smoothly.

We thank our parents for always believing in us and supporting us throughout to reach our goal for this research.

Finally, we thank our faculty members and peers from whom we gathered a lot of knowledge and that helped us to think deeply to accomplish our research.

TABLE OF CONTENTS

LIST OF FIGURES.....	i
LIST OF TABLES.....	ii
CHAPTER 1 Introduction	01
1.1 Objective.....	01
1.2 Methodology	02
1.3 Thesis Outline.....	03
CHAPTER 2 Literature Review	04
2.1 Machine learning	04
2.2 Related Work	05
CHAPTER 3 Feature Selection and Scaling.....	08
3.1 Feature Relevance.....	09
3.2 Feature Redundancy.....	10
3.3 Enumerations of features.....	11
3.4 Feature Ranking.....	17
3.5 User Rating.....	17
3.6 Figures of Sample Resume and Dataset.....	23

CHAPTER 4 Experimental Setup and Analysis.....	26
4.1 Dataset Initialization.....	26
4.2 Algorithms Used	27
4.3 Experimental Results.....	33
4.4 Challenges.....	33
 CHAPTER 5 Conclusions.....	 35
5.1 Conclusions.....	35
5.2 Future Work.....	35
 References.....	 36

LIST OF FIGURES

Figure 3.6.1: Sample CV 1 st page.....	23
Figure 3.6.2: Sample CV 2 nd Page.....	24
Figure 0.6.3. An Example figure.....	25
Figure 4.1.1. Dataset Initialization.....	26
Figure 4.2.2. Dataset Initialization.....	26
Figure 4.3.3. Dataset Initialization.....	27

LIST OF TABLES

Table 3.3.1 Scoring criteria for experience.....	14
Table 3.3.2 Scoring criteria for skills.	15
Table 3.3.3 Skills as displayed in the given resumes.	15
Table 3.5.1 Rating criteria for features.....	18
Table 3.5.2 Rating criteria for academics.....	19
Table 3.5.3 Rating criteria for experience.	19
Table 3.5.4 Rating criteria for publication and training.....	20
Table 3.5.5 Rating criteria for skills.....	21
Table 4.3.1: Accuracies Obtained.....	29

CHAPTER 1

Introduction

The purpose of this research is to compare the performances of a system using machine learning algorithms that will provide with a short list of recommended resumes from the large number of resumes that will be given as input. However, any of the machine learning based systems make their decisions based on some strong assumptions, something they discern from the well-structured and standardized previous data of the similar field that they have been provided with. Similarly, in the early phase of our research we will be providing our system with an ample amount of training data, about 500-600 evaluated resumes, which will make it acquire relevant knowledge based on which it will be making the recommendations. Our system will sort and search the training data and it will correlate with the test results for checking the certitude of its preliminary assumptions. In accordance with our working plan, so far we have developed a specific model containing different functional blocks of the system and their sub blocks within them. We have implemented and gotten results from different algorithms giving different accuracies which help us determine which algorithm to best use when using the system.

1.1 Objective

Recruiting is an important function in the process of human resource management, as labor is seen as the more important ones from the factors of production. The recruiting of the appropriate person is a challenge faced by most companies, as well as the unavailability of certain candidates in some skill areas has long been identified as a major obstacle to companies' success [5]. With the increasing amount of information, organizations need to evolve their way of handling the recruitment process. Even though recruitment selection processes vary from

country to country or even organization to organization, there are similarities as well. Suppose, when choosing a professor, a university would recruit someone who was well-established in his field academically, which would mean, putting more emphasis on his academics, publications, number of years taught more than his extra-curricular activities. This method of selection is usually commonplace practice for many jobs. This research idea is conducted in the context of Bangladesh, the problem arises when the recruiting process gets a little skewed due to societal stereotypes, an example would be a good CGPA would make a good employee regardless the job. Although it has decreased somewhat, the problem still persists in the form of the initial selection process starting with evaluating a single factor rather than the multitude of attributes present in the candidates resume. This ends up in resulting the lesser qualified candidates to be chosen or sometimes the wrong ones. The system we plan to further develop is to give everyone a fair chance to be chosen for the right job. To achieve this we have come up with a method of scoring that evaluates each of the attributes according to a scale tailored to the organizations requirements. After that the information will be evaluated by the automated system that uses supervised machine learning algorithms to select or shortlist the best amongst the candidates.

1.2 Methodology

The method of selection, usually commonplace practice for many organizations, is very biased, especially in Bangladesh. We are trying to make the selection process as unbiased as possible. We can achieve this by treating the different sections of the resumes like experience, academics, training, etc. as a separate attribute. This way it would be easier to scale and score them using machine learning algorithms, which would be trained by data from previously employed candidates of the similar jobs. Supervised machine learning algorithms like decision tree, SVM,

multi-linear regression and Bayesian Ridge regression are used to teach the system and increase learning accuracy.

1.3 Thesis Outline

This thesis paper is divided into five chapters. The outline of each chapter is given below.

Chapter 1 introduces our topic; describe the objective, i.e. our motivation behind our work and the methodology used.

Chapter 2 presents the relevant works we have studied to come up with our own model.

Chapter 3 describes the feature extraction, selection, scaling and how we used that to create the basis for our model.

Chapter 4 discusses results and comparison of the algorithms we used.

Chapter 5 is the conclusion and future work. Here we summarize our idea and provide the scope for improvement in the future.

CHAPTER 2

Literature Review

In this chapter we discuss the research we have done, and give an overview of the tools we used.

2.1 Machine Learning

Machine learning is about learning from data and making predictions and/or decisions. Usually we categorize machine learning as supervised, unsupervised, and reinforcement learning. In supervised learning, there are labeled data; in unsupervised learning, there are no labeled data; and in reinforcement learning, there are evaluative feedbacks, but no supervised signals. Classification and regression are two types of supervised learning problems, with categorical and numerical outputs respectively. Machine learning is based on probability theory and statistics [25] and optimization [26], is the basis for big data, data science, data mining, information retrieval, etc. and becomes a critical ingredient for computer vision, natural language processing, robotics, etc.

Machine learning is a subset of artificial intelligence (AI), and is evolving to be critical for all fields of AI. A machine learning algorithm is composed of a dataset, a cost/loss function, an optimization procedure, and a model [27]. A dataset is divided into non-overlapping training, validation, and testing subsets. A cost/loss function measures the model performance, e.g., with respect to accuracy, like mean square error in regression and classification error rate. Training error measures the error on the training data, minimizing which is an optimization problem. Generalization error, or test error, measures the error on new input data, which differentiates machine learning from optimization. A machine learning algorithm tries to make the training error, and the gap between training error and testing error small. A model is under-fitting if it cannot achieve a low training error; a model is

over-fitting if the gap between training error and test error is large. A model's capacity measures the range of functions it can fit.

2.2 Related Work

Through our long study about similar type of research works that have been done by now, we have learnt that there have been a significant amount of recommendation systems been built regarding different topics and problems. In this section, we will be trying to highlight some of those works and techniques that have been used and which are some way related to our work.

First of all, as the measure of information grows, a recommendation framework is important to help coordinate the right applicant with the correct job. To do as such, recommendation methods, for example, content-based sifting and collaborative filtering also crossover methodologies can be applied [13].

Collaborative filtering has been one of the most effective methods regarding making certain recommendations. Through collaborative filtering the system recommends user products of similar sort that they like some time before. Interaction based recommendation systems can be some good examples of systems that use collaborative filtering [15]. Collaborative filtering based systems collect user ratings about different items of certain field. While recommending a certain item collaborative filtering systems prioritize other user's opinions too. For example in case of movie recommendation collaborative filtering systems try to find out similar type of users that are preferring movies that are closely related or of similar genres, then whenever they encounter a user of that same type, it recommends movies that have been liked by the same kind of users previously. There remains a big number of techniques associated with collaborative filtering, but we would like to mention Memory based Approaches and Model based Approaches [17].

Recommendation systems that use the Memory based collaborative filtering techniques analyze the users and all the items continuously for being able to make the perfect recommendation with better accuracy. These can be classified into collaborative filtering, content based recommendation and hybrid systems.

On the other hand, Model based approaches uses some theoretical models that represent the behavior of user rating. It does not use raw data for the prediction purpose. Rather some model parameters are formed from the user rating dataset, which the model uses for making the predictions. For example, two of the model based approaches are Clustering and Bayesian Network which lead to some very accurate predictions which really scalable and perfect [19].

Moreover, when collaborative filtering systems make their recommendations mostly from what they have learnt from the responses of a group of similar users, content based systems concentrate mostly on individual item information and omit opinions of other users [18]. Content based approaches that have been used for similar kind of machine learning analysis used to be dependent on keyword search mostly. Though later on those got more upgraded and many statistical methodologies have been introduced [14].

In the context of resume recommendation the name of hybrid systems should also be mentioned. Hybrid system actually combines techniques from both collaborative filtering and content based recommendations [18]. Hybrid systems consider the choice preferences from both sides' employers and the job seekers. It recommends whenever it finds information from both side that are compatible with each other's preferences [16].

In some of the job recommendation systems, case based recommendation is used. In these case based recommendations some set of features describe a specific job where those features contain some well-defined values. The collection of jobs is considered to be the case base where the jobs are represented by different features. Through employing different similarity assessment measures the well-structured jobs are compared with the many different user profiles. Whenever any of the user profiles match with some jobs, those jobs get recommended [20].

CHAPTER 3

Feature Selection and Scaling

Feature selection and proper scaling has been a very crucial factor in terms of systems that function based on feature based machine learning algorithms. The possibility of getting the most desired result mostly depends on choosing the right and most relevant features or attributes in regards to the problem domain. Moreover, feature selection is been a key factor when it comes to working with huge datasets for different research purposes, it does not matter whatever the domain it is in [1].

The mathematical operations and algorithms are run on the matrices that are formed from the features or attributes of certain objects. From which the predictions get generated. So, the result or final prediction the system provides us with is directly related to the selection of relevant attributes. Relevance among the features that have been selected is very important. In this regard, we should clear out exactly what the term relevance refers us to. Relevance among features mean that, if one of the features in a certain space is altered that will bring significant change in the whole classification process including many other instances in that system [2]. If it does not do so, that means that specific feature or attribute is not closely related to that problem domain or we can say it is not relevant. Including such irrelevant features to the system ends up with unnecessary feature redundancy and further complexity [4]. We will be discussing about feature relevance and feature redundancy in details later in this chapter.

Just selecting a number of good features does not necessarily provides assurance of a good result, proper enumeration and scaling of those features also hold huge

significance. Proper enumeration of the selected features is also crucial. This is one of the key parts of the preprocessing phase of working with any machine learning algorithm using a large dataset. Once we standardize the features, the next step is to weight those features. Then we look forward to revising those weights by balancing among them and create the training and test datasets [3]. Fixing the perfect weights for different features is a very sensitive task to be done. We need to maintain a certain balance among the numerical values we assign to each of the features. Large difference in weights of different features can confuse the algorithm and may end up in a bad output. Following a fixed range of numbers for weighting the features. Moreover, while measuring the numerical values for different features, ranking and discriminating power of different features has to be fixed either [5].

3.1 Feature Relevance

Selection of features that are relevant with the problem domain directly contributes to the final result that gets produced by the system. Moreover, selected features have to be have enough relevant with each other also so that changing one feature causes significant change in the effect of another one. This relevance and relation between optimal features is very crucially important for any system that works on feature based concept. Most importantly, decision tree algorithms like ID3 [7], C4.5 [8] or CART [6] these seemed to provide bad output with datasets that are formed with irrelevant features.

Some of the definitions of feature relevance are based on assumptions that each and every features are Boolean so that their remains no noise [9]. According to that, suppose we have dataset D where there features are $\theta_1, \theta_2, \theta_3, \theta_4, \dots \dots \dots \theta_n$

and the system works on concept C. Now, a feature θ_1 can be called relevant to that concept if and only if that feature θ_1 appears in each and every Boolean formula that is representing the concept C.

It can be defined in another way that allows the feature to have noise and to be of multi-valued. Such definitions assume that the values of these multi-valued features are varying systematically and they have some category membership [10]. According to that, a feature θ_1 can be relevant if and only if there exists a θ_2 and y such that $C(\theta_1 = \theta_2) > 0$ and $C(Y = y \mid \theta_1 = \theta_2) \neq C(Y=y)$. This definition states that θ_1 can be called relevant if changing its value can change the estimates for the class Y.

3.2 Feature Redundancy

The term feature redundancy refers to the concept of feature correlation. Two features θ_1 and θ_2 can be called two of the redundant features if their numerical values are totally correlated to each other [11].

In this regard, first of all we should introduce the term Markov Blanket. Suppose we have a feature θ_i and suppose $M_i \subset \theta_i \mid i \in M_i$. Here M_i is known to be a Markov Blanket if and only if

$$P(\theta - M_i - \{\theta_i\}, C \mid \theta_i, M_i) = P(\theta - M_i - \{\theta_i\}, C \mid M_i) \dots \dots \dots (1)$$

The Markov Blanket M_i assumes the information of the feature θ_i about the concept C. Not only the feature θ_i , has it assumed the information of all other features too [12]. Now to define feature redundancy we can say that suppose we

have a set of feature called S , a feature in it can be called redundant if and only if it is weakly relevant and has a Markov Blanket M_i within that set of feature S .

3.3 Enumerations of features

While preparing the dataset for a system on which different machine learning algorithms will be run the matter of enumeration of the features plays a key role. Machine learning algorithms like SVM can provide very bad result if it finds too large differences among the feature values in the dataset. Because while minimizing the cost function through gradient descent the large differences among features make it very difficult for the algorithm to reach the best possible result. The gradient takes very long time to get the optimal solution and in fact in most cases it does not reach that solution. That is why as part of the preprocessing phase it is very important to fix a certain range of number values that will be used for the enumeration of features.

For this system of ours we enumerated the selected features in such a way so that all the features are getting weights which are similar to each other. We fixed the numbers from the range -1 to 1 to be allocated to the features. Which will ensure each of the features are getting values that are essentially in the same range of values with the other features. Now while the minimizing the cost function the gradient will be able to have much more straight path to look for the optimal result.

A feature with a numerical value of 1 means it is in its maximum value, while -1 to refer to the minimum. The steps of increasing and decreasing the value are kept variable regarding specific features and their rank in respect to the concept of the problem. We considered to take the floor value of a certain feature if it tends to go beyond the ceiling that is fixed to 1. Unavailability of any information about a

certain feature will make the numerical value of that feature to be 2 which is referred to the system as null.

As we designed the system considering the recruitment of university teachers, so the features we have selected are closely related with the academic domain. We selected ten major features to be used for generating the prediction which are academics, CGPA, experience, training, professional certificate, publications, skill, English language proficiency, extracurricular interests and availability.

3.3.1 Academics

The feature academics is scored from the institutional background and degrees that a candidate has earned. We have considered the rankings of the institutions that the candidates got the degrees from. For this we used the official ranking of universities provided by University Grant Commission Bangladesh.

For each of the post graduate level degrees we added 0.3 to the feature value. In case of post graduate degrees we have omitted institutional ranking for avoiding further complexity for this prototype system. But while scoring bachelor degrees the score is measured according to the official university ranking provided by the University Grants Commission Bangladesh. If the candidate has completed his or her bachelor degree from a university ranked from 1 to 5 that person gets a score of 0.5 out of 1 in his academics feature. If the ranking is 6 to 10, 10 to 20 they get 0.4 and 0.3 subsequently. In case of ranking below 20 the score is 0.2. Candidates with a master's degree gets 0.3 added to the score of their bachelor degree, which is allocated score for masters. The total becomes the final score for the academics feature.

For further illustration we would like to explain with an example. Suppose, a candidate has completed his bachelor degree from BRAC University and he also has a master's degree. Now from University Grants Commission Bangladesh we find that BRAC University has a ranking of fourth in Bangladesh that means the candidate is getting full 0.5 score for his bachelor degree. Moreover, as he also has masters 0.3 will get added to that 0.5. So,

The value of his academics feature = Score from Bachelor degree + Score from Master's Degree

$$= 0.5 + 0.3 = 0.8 \text{ (out of 1).....(2)}$$

The final score of the academics feature for that candidate is 0.8 out of 1.0 If he did not have any master's degree the score would be only 0.5, then that extra 0.3 would not get added.

3.3.2 Cumulative Grade Point Average

Scoring the CGPA is much simpler. As maximum CGPA possible is 4.00 we considered that to be 1 under our scoring range, which is the maximum. Any other CGPA in the scale of 4 is converted to our scale of 1.

For instance, a candidate with CGPA 4.00 gets maximum score of 1 added to his CGPA feature. If the CGPA was 3.5 then the score would be,

$$\text{Score} = 3.5 / 4 = 0.875$$

So, here the CGPA 3.5 out of the scale of 4 has got converted to 0.875 which is out of 1.

3.3.3 Experience

For the feature experience there we have created six specific criterion for scoring according to the years of experience a candidate has got. The table below represents that criterion.

According to the above given scale, a fresher candidate gets the minimum score which is -1 according to our range of values. A candidate with experience of more than 8 years gets the maximum score which is 1. Except the score -0.75 for less than 2 years of experience, in case of all other intervals the score changes with a step of 0.5.

Number of years of experience	Score
No experience (Fresher)	-1
Experience < 2	-0.75
2 < experience <= 4	-0.5
4 < experience <= 6	0
6 < experience <= 8	0.5
Experience > 8	1

Table 3.3.1: Scoring criteria for experience

3.3.4 Skill

We have set the scoring policy depending on the types of CVs that we have come across. Candidates that have mentioned about more than three skills, we fixed it to score their skills feature to be maximum that means 1. But, those who have mentioned three or less, then there we considered the degree of their skills in those areas or subjects. Which is given below.

Good	Very good	Excellent
0.1	0.2	0.3

Table 3.3.2: Scoring criteria for skills

According to this criteria, suppose a candidate has mentioned about three skills MS Word, MS Excel and PowerPoint. Consider the table below as an example,

Skill	Good	Very Good	Excellent
MS Word		√	
MS Excel		√	
Power Point			√

Table 3.3.3: Skills as displayed in the given resumes

From the above skills and their degrees mentioned by a certain candidate if we calculate the score,

$$\begin{aligned}\text{Score} &= \text{MS word (Very good)} + \text{MS Excel (Very Good)} + \text{Power Point (Excellent)} \\ &= 0.2 + 0.2 + 0.3 \\ &= 0.7\end{aligned}$$

3.3.5 Publication and Training

In case of the features like training and publications we have considered the number of training programs a particular candidate has attended and the number of publications they have made. We focused mostly on the numbers rather than anything else. For each of the publications that a candidate has made 0.2 gets added to the publications feature. For example, someone with 4 publications get a score of $4 * 0.2 = 0.8$. Just same as that for each of the training programs attended a candidate gets same score 0.2. A candidate with 5 trainings attended gets a score of $5 * 0.2 = 1$.

3.3.6 Language Proficiency

For the enumeration of the feature English language proficiency we considered the IELTS score to be the most standard scale to measure with. An IELTS score of 7 or more is taken as the maximum value 1 for this feature. If it is less than 7, then the feature value gets curbed according to the scale of 1.

For instance, if a candidate has an IELTS score of 7 then it adds 1 to the feature English Proficiency, where 1 is the maximum. If there is some other candidate that has a score of 6.5 then the score is,

$$\text{Score} = 6.5 / 7 = 0.92.$$

3.3.7 Availability and Extracurricular Activities

The feature availability is comparatively simpler to be discussed. It indicates whether the candidate is currently available or is not in any job right at that moment. The scoring is done in a discrete manner. If the candidate is available the score is 1 and if not it is given -1. These are the features that we have found out to be most relevant relating to the problem domain. As we have hardly come across resumes that have mentioned about extracurricular activities in them, so we decided to score it in a discrete way. If there is any candidate that has mentioned about his / her extracurricular activities we have scored with 1, if not then -1.

3.4 Feature Ranking

While building this prototype system we have considered the system will recommend university teachers for a certain university. In this regard the user is the university human resource management department who will be sorting out the resumes. As we are considering a domain that mostly concentrates on academic factors, we have set the feature ranking mostly according to that. The ranking of the features are given below.

- Academics
- Experience
- Publication / Training
- Skills
- Availability
- English Proficiency / Extra-Curricular Activity

3.5 User Rating

Though we have received a number of resumes from BRAC University as part of the research purposes, it was difficult to know the actual policy of how the university rate the candidates through the resumes and choose the perfect candidate because of confidentiality issues. That is why we fixed a certain experimental policy of choosing the candidate and made a rule of how the users may rate different resumes.

For this user rating purpose we have fixed a scale of 5. That means the user will score a certain candidate out of the number 5 based on that candidate's different credentials. These 5 points are divided among all the features, where the percentage of the total rating point a feature gets depends on its ranking among all the other features. Score allocated according to the feature ranking is given with Table 3.5.1.

Feature Label	Points allocated (out of 5)
Academics	2
Experience	1
Publications / Training	0.5
Skills	0.5
English Proficiency	0.25
Availability	0.5
Extra-Curricular Activity	0.25
Total	5

Table 3.5.1: Rating criteria for features

3.5.1 Academics

For the feature academics, 2 out of 5 from user rating point is allocated. In the dataset, if a certain candidate has a score of 0.5 or less as their academics feature value, then the user adds 1 to the user rating point. If it is greater than 0.5 but less than 0.8, 1.5 is added. In case of more than 0.8 the candidate gets full point 2 which is allocated for the feature academics. Table 3.5.2 shows the rules followed.

Scale points	Points awarded (out of 2)
<0.5	1
$0.5 \leq P \leq 0.8$	1.5
>0.8	2

Table 3.5.2: Rating criteria for academics

3.5.2 Experience

The experience feature is allocated the score of 1 out of 5 from the total user rating point. There are five different divisions according to the years of experience a particular candidate has got. The score starts from 0.2 ends at 1 which is the maximum, with a step of 0.2.

For a fresher candidate no score is added to the user rating point but for a candidate that as experience more than 8 years gets the full point 1 added to the user rating.

Experience	Points awarded (out of 1)
Exp<= 2	0.2
2 <Exp<= 4	0.4
4 <Exp<= 6	0.6
6 <Exp<= 8	0.8
Exp> 8	1

Table 3.5.3: Rating criteria for experience

3.5.3 Publication and Training

Publications and Training features got 0.5 out of 5 been allocated to their user rating point. Candidates with one or two publications get 0.2 and 0.4 out 5 added to their user rating point subsequently. Those who have more than three publications get the full 0.5 that is been allocated for this feature added to the user rating point.

Scale Points	Points awarded (out of 0.5)
1 publication (0.2)	0.2
2 publication (0.4)	0.4
3 or more publications	0.5

Table 3.5.4: Rating criteria for publication and training

3.5.4 Skills

Skills feature is allocated with 0.5 of 5 from user rating point. Where the score starts from 0.05 and the maximum is 0.5. If the feature skills in the dataset has a value 1 then the user rating point is maximum which is 0.5. In the opposite way, skills feature with value with lowest 0.1 gets 0.05 added to user rating. The point increases with a step of 0.05. It is shown in Table 3.5.3 in page 21.

Skill Feature Value	Points awarded (out of 0.5)
1	0.5
0.9	0.45
0.8	0.40
0.7	0.35
0.6	0.30
0.5	0.25
0.4	0.20
0.3	0.15
0.2	0.10
0.1	0.05

Table 3.5.5: Rating criteria for skills

3.5.5 Others

English proficiency is allocated with 0.25 out of total 5 of user rating. Here, candidates with an IELTS score of 7 or more gets full of 0.25 to their user rating point. Anybody with a score of less than 7 gets their user rating curbed according to the scale of 0.25. The user rating point of the availability and extracurricular activities feature is scored is given based on some Boolean decision. If a candidate is available on that period of time he gets 0.5 added to the user rating point that is allocated for the availability feature, if not available 0 is added. Just same as that, if someone mentioned about their extracurricular activities in the resume they get 0.25 added which is the maximum that has been allocated for this feature, otherwise they get 0.

3.6 Figures of Sample Resume and Dataset



BRAC University

Academic Qualification:

Level (in descending order)	Area of study	Concentration/ Major	Year of Pass	Name of Institution	Division/ Class/CGPA	Percentage of marks obtained
Bachelor of Science	B.Sc. Engineering	EEE	2014	Chittagong University of Engineering & Technology (CUET)	CGPA:3.66	77.20%
H.S.C		SCIENCE	2009	Moulvibazar Govt. College	GPA:5.00	80%
S.S.C		SCIENCE	2007	The Flowers' kg and high school	GPA:5.00	80%

English proficiency (IELTS/ TOEFL/ others.) Score:

Computer Proficiency (MS Word): ☒ Excellent
Good Very Good

Computer Proficiency (MS Excel): ☒ Excellent
Good Very Good

Computer Proficiency (Others): ☒ Excellent
Good Very Good

Add more (if needed)

Experience Information: ☐ Fresh ☒ Experienced

Experience 1:

Employer:

Designation:

From:

Core Responsibilities:

Address:

Phone:

2

Figure 3.6.1: Sample CV 1st page



BRAC University

Total Year of Experience: Months

Professional Qualification:

Name of Degree:

Institute:

From: To:

No. of Publication (If any): National International

Attach details:

Training:

1. Title with brief Description:

Institute:

From: To:

2. Title with brief Description:

Institute:

From: To:

Figure 3.6.2: Sample CV 2nd Page

From the formatted CVs we were able to prepare the following dataset, as shown in Figure 3.6.3 below, using the scaling and scoring system we discussed in this chapter.

Sample of dataset prepared:

Academic	CGPA	Experience	Training	ProfessionalCe	Publications	Skill	EnglishSkill	ExtraCurricular	Availability	UserRatings
1	0.96	-0.75	0.2	2	0.2	0.7	0.92	2	1	3.48
0.6	0.925	-0.5	0.2	2	0.8	0.6	2	2	-1	2.7
0.8	0.9	-0.75	2	2	2	0.3	2	2	-1	1.85
0.8	0.9125	-1	2	2	2	2	1	2	1	2.45
1	0.82	0	2	2	0.6	1	1	2	-1	3.85
0.8	0.94	-1	2	2	2	0.9	2	2	1	2.65
0.8	0.945	0.5	2	2	2	0.9	1	2	-1	2.9
0.8	0.955	-1	0.8	2	2	0.9	2	2	1	2.65
1	0.56	1	2	1	2	0.6	2	2	1	4
0.8	0.93	-0.75	2	2	2	0.4	2	2	-1	1.9
0.8	0.955	-0.5	2	2	0.6	0.9	2	2	-1	2.85
1	0.75	1	0.4	1	1	1	2	2	-1	4
1	0.75	0.5	2	1	1	2	2	2	-1	3.3
0.8	0.9775	0	0.2	2	1	0.9	1	2	-1	3.3
1	1	1	1	1	1	2	2	2	-1	3.5
1	1	1	0.4	1	1	2	2	2	-1	3.5
0.8	0.9375	-0.75	2	2	2	0.6	2	2	1	2.5
0.8	0.8275	0	0.8	2	2	2	2	2	-1	2.1
0.8	0.89	-0.75	0.2	2	2	2	2	2	1	2.2
0.5	0.86	-0.75	2	1	2	0.3	0.785	2	1	2.55
0.6	0.8475	0	1	2	2	0.9	2	2	-1	2.35
0.5	0.925	-1	0.2	2	2	1	2	1	1	2.95
0.9	0.9	-0.75	2	2	2	0.9	0.928	2	1	3.39
0.8	0.8575	-0.75	2	2	2	0.7	2	2	-1	2.05
0.5	0.9125	0	0.2	2	2	0.9	2	2	-1	2.65
0.8	0.8625	-0.75	1	1	2	2	2	2	-1	1.7
0.6	0.6375	-0.75	0.2	1	2	0.8	0.928	2	1	2.84
0.8	0.895	-0.75	0.2	2	2	0.8	2	2	1	2.6
0.8	0.7475	0	0.4	2	2	0.8	2	2	-1	2.6
0.8	0.7875	0	2	2	2	1	2	2	-1	2.7
0.8	0.88	-1	0.2	2	2	0.4	2	2	1	2.4

Figure 3.6.3. An Example figure.

CHAPTER 4

Experimental Setup and Analysis

4.1 Dataset Initialization

First we setup the initial training and testing data as shown in the figure below

```
#Read training dataset
data = pd.read_csv("E:\Library\study\University\BRAC\Semesters\Thesis\CV\codes and results\data12.csv")
df = pd.DataFrame(data)

x_train = df[['Academics', 'CGPA', 'Experience', 'Training', 'ProfessionalCertificate', 'Publications', 'Skill',
              'EnglishSkill( IELTS )', 'ExtraCurricular', 'Availability']]
y_train = df[['UserRatings']]

#reading test data
data1 = pd.read_csv("E:\Library\study\University\BRAC\Semesters\Thesis\CV\codes and results\dtestsadman1.csv")
df1 = pd.DataFrame(data1)

x_check = df1[['Academics', 'CGPA', 'Experience', 'Training', 'ProfessionalCertificate', 'Publications', 'Skill',
                'EnglishSkill( IELTS )', 'ExtraCurricular', 'Availability']]
y_check = df1[['UserRatings']]
```

Figure 4.4.1. Dataset Initialization.

Then we train the data sets using the four algorithms we chose,

```
#Training the system using training
#using Decision tree
clf = tree.DecisionTreeRegressor()
clf.fit(x_train, y_train, sample_weight=None)

#using SVM
clf = svm.SVR()
clf.fit(x_train, y_train)

#using Bayesian Ridge
clf = linear_model.BayesianRidge()
clf.fit(x_train, y_train)

# Using multi linear regression
clf = LinearRegression()
clf.fit(x_train, y_train, sample_weight=None)
```

Figure 4.5.2. Dataset Initialization.

After that we predict the outcomes,

```
#predicting outcome based on training and tesing dataset
prediction = clf.predict(x_check)
print(prediction)

#calculating accuracy
print clf.score(x_check, y_check) * 100 , "%"
```

Figure 4.6.3. Dataset Initialization.

4.2 Algorithms Used

4.2.1 Decision tree:

Decision trees are a part of supervised learning algorithms. It can be used to solve both regression and classification problems. A decision tree is a graphical representation of possible solutions to a decision based on certain conditions:

- a. Place the best attribute of the dataset at the root of the tree.
- b. Split the training set into subsets. Subsets should be made in such a way that each subset contains data with the same value for an attribute.
- c. Repeat steps a and b until you find the leaf node of all the branches.

Algorithm:

```
def decisionTreeLearning(examples, attributes, parent_examples):
    if len(examples) == 0:
        return pluralityValue(parent_examples)
    # return most probable answer as there is no training data left
    elif len(attributes) == 0:
        return pluralityValue(examples)
    elif (all examples classify the same):
        return their classification
```

```

A = max(attributes, key(a)=importance(a, examples)
# choose the most promising attribute to condition on
tree = new Tree(root=A)
for value in A.values():
    exs = examples[e.A == value]
    subtree = decisionTreeLearning(exs, attributes.remove(A), examples)
    # note implementation should probably wrap the trivial case returns into trees
for consistency
    tree.addSubtreeAsBranch(subtree, label=(A, value))
return tree

```

Advantages:

- a. Decision trees require relatively little effort from users for data preparation.
- b. Nonlinear relationships between parameters do not affect tree performance.

Disadvantages:

- a. High Variance - The prediction model gets unstable with a very small variance in data.
- b. A highly complicated Decision tree tends to have a low bias which makes it difficult for the model to work with new data

4.2.2 SVM:

A Support Vector Machine (SVM) is a supervised machine learning algorithm that can be employed for both classification and regression purposes. It is based on the idea of finding a hyperplane that converts a dataset into two classes. Hyperplane can be refer to a line that linearly separates and classifies a set of data and Support vectors are the data points nearest to the hyperplane. Distance between the hyperplane and the nearest data point from either set is known as the margin. The aim is to choose a hyperplane with the greatest possible margin between the

hyperplane and any point within the training set, giving a greater chance of new data being classified correctly.

Algorithm:

add (xc, yc) at the trainingset

set qc = 0

compute f(xc) and h(xc)

if ($|h(xc)| < e$)

add newsample to the remainingset and exit

compute h(xi), $i=1..l$

while (newsample is not added into a set)

update the values b and g

find least variations (lc1, lc2, ls, le, lr)

find min variation dqc = min(lc1, lc2, ls, le, lr)

let flag the case number that determinates dqc

(lc1=1, lc2=2, ls=3, le=4, lr=5)

let xl the sample that determines dqc

update qc, qi, $i=1..l$ and b

update h(xi), $i \in \{1..l\}$

switch flag

(flag = 1)

add newsample to supportset

add newsample to r matrix

exit

(flag = 2)

add newsample to errorset

exit

(flag = 3)

```

if (ql = 0)
    move sample l from support to remainingset
    remove sample l from r matrix
else [ql = |c|]
    move sample l from support to errorset
    remove sample l from r matrix
(flag = 4)
    move sample l from error to support
    add sample l to r matrix
(flag = 5)
    move sample l from remaining to supportset
    add sample l to r matrix

```

Advantages:

- a. It has a regularization parameter, which makes the user think about avoiding over-fitting.
- b. It uses the kernel trick, so you can build in expert knowledge about the problem via engineering the kernel.

Disadvantages:

- a. Picking a good kernel function is difficult.
- b. Long training time on large data sets.

4.2.3 Multi-Linear Regression:

Multiple linear regression uses many input variables and the create relationships among them. In multiple linear regression, not only the input variables are potentially related to the output variable, they are also related to each other, this is referred to as multi co-linearity.

The model for the multiple linear regression,

$$f(X) = a + (B_1 * X_1) + (B_2 * X_2) \dots + (B_p * X_p) \dots \dots \dots (4)$$

Here, input variables = X,

Specific input variable= X_p ,

Coefficient (slope) of the input variable, $X_p = B_p$,

And intercept= a

In, Simple linear regression it uses one-to-one relationship between the input and output variable. But in multiple linear regression, there is a many-to-one relationship as it uses multiple input variable.

Algorithm:

Got a bunch of points in R^2 , $\{(x_i, y_i)\}$.

Want to fit a line $y = ax + b$ that describes the trend.

We define a cost function that computes the total squared error of our predictions w.r.t. observed values y_i $J(a, b) = \sum (ax_i + b - y_i)^2$ that we want to minimize.

See it as a function of a and b: compute both derivatives, force them equal to zero, and solve for a and b.

The coefficients you get give you the minimum squared error.

Can do this for specific points, or in general and find the formulas.

Advantages:

- a. Ability to determine the relative influence of one or more predictor variables to the criterion value.
- b. The ability to identify outliers, or anomalies.

Disadvantages:

- a. At the most basic level, like most other methods of multivariate statistical analysis MR works by rendering the cases invisible, treating them simply as the source of a set of empirical observations on dependent and independent variables

b. Multi co-linearity, i.e. the danger of including several terms in the equation that are well correlated is that we will effectively be using the same x variance more than once to explain variance in y.

4.2.4 Bayesian Ridge:

Bayesian ridge is used to estimate a probabilistic model of the regression problem. The prior for the parameter w is given by a spherical Gaussian:

$$p(w|\lambda) = \mathcal{N}(w|0, \lambda^{-1} \mathbf{I}_p) \dots\dots\dots(5)$$

The conjugate prior for the precision of the Gaussian over α and λ are chosen to be gamma distributions. The resulting model is called *Bayesian Ridge Regression*, and is similar to the classical ridge. The parameters w , α and λ are estimated jointly in the time of fitting of the model. The remaining hyper parameters are the parameters of the gamma priors over α and λ .

Algorithm is similar to linear regression, with the exception of inclusion of a prior/posterior probability.

Advantages:

- a. Bayesian learning involves specifying a prior and integration, two activities which seem to be universally useful.
- b. Bayesian inferences require skills to translate subjective prior beliefs into a mathematically formulated prior. If you do not proceed with caution, you can generate misleading results.

Disadvantages:

- a. It is not a separate algorithm but a statistical approach with inference.
- b. It provides a convenient setting for a wide range of models, such as hierarchical models and missing data problems.

4.3 Experimental Results

Using the algorithms discussed in the previous section, we acquired the results in the Table 4.3.1 below

From table below we can deduce that SVM and Decision Tree algorithms provide the best results. Although the Decision Tree give a higher percentage than the SVM algorithm, we would recommend SVM. This is due to the reason that a decision tree starts the process of building a tree from scratch every time the algorithm is called but with a different root node and hence gives more volatile results as well as being more prone to over-fitting as the complexity of the dataset increases.

Algorithms	Decision Tree	SVM	Bayesian Ridge	Multi-Linear
Accuracy	98.787%	90.592%	74.875%	74.943%

Table 4.3.1: Accuracies Obtained.

4.4 Challenges

The idea of data analysis and machine learning is quite new. Most work done on machine learning and data analysis out there have created their own dataset, algorithm, formatting data of their own. There is no generalized rule or formal criteria for the process. During our thesis we faced some challenges. First of all, as our thesis is on “Performance Analysis of Machine Learning Algorithms in Résumé Recommendation” so we need a large number of resumes. But to get this amount of resumes we faced lots of problems because most of organizations denied to give us their employee resume for research due to confidentiality clauses as some organizations maintain strong confidentiality about their employee’s information. So to get these 500-600 resumes we maintain so many criteria to

ensure confidentiality of this resumes that we use this only on research purpose. Secondly, the resume we got from several organizations were not formatted data and scaling of that unformatted data was very challenging as there is no generalize scale or formatting process in this type of research so we have to build our own scale and data format. According to scale and data format we build our own data set which is a very time consuming and work of patience.

CHAPTER 5

Conclusions

5.1 Conclusion

To conclude, we discussed about a model we planned to develop for a resume recommendation system and the performance analysis of the algorithm. Along with that we explained how a resume system works, its function, algorithms, dataset etc. We researched on some of the existing algorithms by which we got the best possible outcomes. Using the scoring of resumes we came up with, the recommendation system which can be used for any kind of job recruitment process, as the organization will be able to alter the scale to their needs. Our main purpose was to find the best algorithm to establish a prediction of the best solution from the accuracies provided using data analysis and machine learning approach based on our dataset, the model we created, i.e. the scoring system and existing algorithms.

5.2 Future Work

In our thesis, we presented an evaluation of machine learning algorithms on a model prepared by us for improving the recruitment processes of organizations. At this point of our thesis, we analyzed some popular algorithms and ran simulations on our dataset so that we can recommend the best algorithm to use. We manually took inputs from resume and trained the machine using the algorithms. In the future, we want to build a system that can automatically take input from resume, by means of a formatted resume and train itself. Moreover we have interest in a hybrid recommendation system which will have more function and features such as weighted, switching, mixed, feature combination, cascade, feature augmentation, and model which can perform better than other approaches.

REFERENCES

- [1] Guyon, I. and Elisseeff, A. (2003). An introduction to variable and feature selection. *Journal of machine learning research*, 3(Mar), 1157
- [2] Blum, A. L. and Langley, P. (1997). Selection of relevant features and examples in machine learning. *Artificial intelligence*, 97(1-2), 245-271.
- [3] Kuncz, J. and Chatterjee, S. A MACHINE-LEARNING APPROACH TO PARAMETER ESTIMATION.
- [4] Zeng, X. and Martinez, T. R. (2004, July). Feature weighting using neural networks. In *Neural Networks, 2004. Proceedings. 2004 IEEE International Joint Conference on* (Vol. 2, pp. 1327-1330). IEEE.
- [5] Shankar, S. and Karypis, G. (2000, August). A feature weight adjustment algorithm for document categorization. In *KDD-2000 Workshop on Text Mining*, Boston, USA.
- [6] L. Breiman, J.H. Friedman, R.A. Olshen and C.J. Stone, *Classification and Regression Trees* (Wadsworth, Belmont, CA, 1984).
- [7] J.R. Quinlan, Induction of decision trees, *Machine Learning* 1 (1986) 81-106; reprinted in: J.W. Shavlik and T.G. Dietterich, eds., *Readings in Machine Learning* (Morgan Kaufmann, San Mateo, CA, 1986).
- [8] 1981 J.R. Quinlan, *C4.5: Programs for Machine Learning* (Morgan Kaufmann, San Mateo, CA, 1993).
- [9] H. Almuallim and T. G. Dietterich, *Efficient Algorithms for Identifying Relevant Features*. Orgeon, 2014.
- [10] G. H. John, R. Kohavi and K. Pfleger, "Irrelevant Features and the Subset Selection Problem", presented at the Conf. of Machine Learning Proceedings of the Eleventh International Conference, shers San Francisco, USA

- [11] L. Yu and H. Liu. Efficient Feature Selection via Analysis of Relevance and Redundancy.
- [12] S. Fu, M. Desmarais and W. Chen, Reliability Analysis of Markov Blanket Learning Algorithms. IEEE, 2010.
- [13] Sovren Group. 2006. Overview of the Sovren Semantic Matching Engine And Comparison to Traditional Keyword Search Engines. Sovren Group, Inc.
- [14] Rafter, R., Bradley, K., and Smyth, B. 2004. Automated collaborative filtering applications for online recruitment services. In Proceedings of the International Conference on Adaptive Hypermedia and Adaptive Web-based Systems. 1892, (Trento, Italy, August. 2000). AH '00, 000, 363-368. DOI: 10.1007/3-540-44595-1_48.
- [15] Malinowski, J., Keim, T., Wendt, O., Weitzel, T. 2006. Matching People and Jobs: A Bilateral Recommendation Approach. In Proceedings of the 39th Annual Hawaii International Conference on System Sciences. 6 (Hawaii, USA, Jan. 4-7, 2006). HICSS '06, 137c.
- [16] Meenakshi Sharma and Sandeep Mann, "A Survey of Recommender Systems: Approaches and Limitations", ICAECE2013.
- [17] B. Bhatt, P. J. Patel and H. Gaudani(2014). A Review Paper on Machine Learning Based Recommendation System. *IJEDR*. [Online]. Available: <https://www.ijedr.org/papers/IJEDR1404092.pdf>
- [18] Zunping Cheng, Neil Hurley," Effective Diverse and Obfuscated Attacks on Model-based Recommender Systems" 2009 ACM
- [19] X. Guo, H. Jerbi and M. P. O'Mahony, *An Analysis Framework for Content-based Job Recommendation*. Dublin, Insight Centre for Data Analytics.
- [20] I. Portugal, P. Alencar and D. Cowan. The Use of Machine Learning Algorithms in Recommender Systems: A Systematic Review, 2015.

- [21] S. T. Al-Otaibi¹ and M. Ykhlef. A survey of job recommender systems, 2012.
- [22] L. Kotthoff, I. P. Gent and I. Miguel. An Evaluation of Machine Learning in Algorithm Selection for Search Problems, 2011.
- [23] J. Brownlee (2013). A Tour of Machine Learning Algorithms. *Machine Learning Mastery*. [Online]. Available: <http://machinelearningmastery.com/a-tour-of-machine-learning-algorithms>
- [24] A. Ng, ‘Features and Polynomial Regression’. *Coursera*, 2011. [Online]. Available: <https://www.coursera.org/learn/machinelearning/lecture/Rqgfz/features-and-polynomial-regression>
- [25] H. Trevor J, T. R. John and F. Jerome H, *Elements of Statistical Learning: data mining, inference, and prediction*, 2nd edition. New York: Springer, 2009.
- [26] S. P. Boyd, L. Vandenberghe, *Convex Optimization*. Cambridge University Press, 2004.
- [27] I. Goodfellow and Y. Bengio and A. Courville, *Deep Learning*. MIT Press, 2016.