Learning Models for Personalized Educational Feedback and Job Selection

Vinay Shashidhar VINAY.SHASHIDHAR@ASPIRINGMINDS.COM
Aspiring Minds

Shashank Srikant SHASHANK.SRIKANT@ASPIRINGMINDS.COM
Aspiring Minds

Varun Aggarwal VARUN@ASPIRINGMINDS.COM
Aspiring Minds

Abstract

Every year millions of students enter the job market in search of employment opportunities. Multiple studies show that these students do not have skills commensurate to the job requirements in industry. They have no feedback on their skill gap with respect to jobs in the marker and steps on how to improve. Also, learners have no easy way to signal their employability i.e. their job suitability to corporates thus making the employment market inefficient. There is a need to objectively quantify employability for different job profiles, to enable employability feedback to learners and also facilitate an easy way to connect meritorious students with matching job opportunities. We propose a new class of models to predict employability as a function of test scores. These models satisfy the constraints of the problem space, which are coordinate wise monotonicity, simplicity and human interpretability. Learning these models require non convex optimization. To address the same, we use particle swarm optimization, a population based optimization method, to learn multiple trade off models. Through a case study we show that the modeling approach is useful to predict employability for the software engineering role, does better than extant models in hiring accuracy and provides new insight in what constitutes employability for the software engineering profile in the IT services industry. We believe that our modeling language and technique has the potential to drive greater meritocracy into the job markets.

Proceedings of the 32^{nd} International Conference on Machine Learning, Lille, France, 2015. JMLR: W&CP volume 37.

1. Introduction

Many million candidates enter the job market every year after completing higher education, vocational education and more recently, through MOOCs (Josh Bersin, 2013). There has been tremendous concern about the lack of employability (job suitability) of these individuals (Economist, 2014)(Kantrow, 2014)(NER, 2014) , the lack of a mechanism for them to easily signal their employability to corporations (Terviö, 2009)(Aspiring-Minds, 2013) and also the absence of feedback on what skills they lack. Likewise, corporations find it hard to sift through the large number of candidate profiles to find those appropriate for jobs. This makes the labor markets, a primary driver for economic growth, inefficient (Naveh et al., 2007).

There is considerable evidence that assessments of cognitive skills, personality, language and knowledge predict job success and are worthy parameters to determine job suitability (Schmidt & Hunter, 1992)(Chang & Xi, 2009). We are interested in using scores of standardized assessment in these sills to predict job suitability. This shall server the following purposes: a) determine the combination of skills needed for various jobs in the market, b) provide feedback to candidates on their job suitability, gaps in their skill set for particular jobs and ways for them to improve upon, c) provide job credentials to candidates to signal employability and d) provide an easy way for companies to filter high quality candidates and provide interview opportunities to them.

We build classification models which can ascertain employability for different jobs based on the assessment scores of an individual. To find what skills are required for a job, the industrial organization psychology (IOP) community conducts *criterion validity studies* (Barrick & Mount, 1991): they deliver the assessments on a current sample of employees, get job performance data on these employees through the company/managers and build predictive models linking assessment scores to job performance. Traditionally, people have used linear regression and second order models to find what parameters correlate to job success, by how much they correlate and what the total predictive ability is (determined by *r*, the Pearson coefficient) (Schmidt & Hunter, 1992). There is a formidable body of literature suggesting what skills among personality traits, language, cognitive skills and domain skills are needed for which jobs (Hunter, 1986)(Lievens & Sackett, 2012)(Sternberg & Wagner, 1993). When developing selection criteria to offer an interview opportunity based on test scores, a classification problem, the IOP community has generally used cut-scores (Cascio et al., 1988), individual thresholds (cut-offs) on various assessment scores. There has also been a tradition of combining cut-scores into linear models using expert judgment. At best, one sees the use of linear models arrived at by the use of logistic regression. There have been a few instances where neural networks and SVMs too were used, but these models haven't scaled due to the issues discussed below (Li et al., 2008).

Machine learning models need to follow certain design constraints to become amenable for the stated purpose. First, they need to be theoretically plausible. For instance, a candidate with a higher score cannot be rejected in comparison to one with a lower score. We have identified coordinate-wise monotonicity (CM, details in §2) as the weakest structure needed for this to happen, a theme also discussed in (Lan et al., 2014).[1] Second, they need to be simple and human interpretable. This is important because we cannot completely depend on data given its non-causal nature and the sample bias it may contain. In addition to providing feedback to students, interpretable models provide considerable of flexibility to HR personnel by giving them an ability to manually tweak the models to favor their domain knowledge and also work with the legal regulations around assessment based selection (Cascio et al., 1988). As an added advantage, it helps discover new insights related to the assessments , as also demonstrated in this paper. Lastly, there is a need for trade-off models where one could choose between models with different type-1 and type-2 errors allowing HR personnel to accommodate prevalent market conditions and allowing them to suit their companies' standards.

In this paper, we propose a class of models intuitive to practitioners in the domain. They are simple and human interpretable and hypothesized to fir the structure of the domain. Given that the optimization of misclassification function for our class of models isn't convex, we propose a heuristic algorithm to do the optimization and provide trade-off models. To the best of our knowledge, these models and algorithms to learn them haven't been explored by the machine learning community before. We finally demonstrate through a case study, how these models are superior to simple linear models, lead to discovery of new knowledge and help both candidates and companies in an efficient skill-identification process.

Specifically, the paper makes the following contribution -

- We propose an ensemble of coordinate-wise monotonic, simple and interpretable classification models for the problem of predicting employability of candidates.
- We demonstrate the use of an evolutionary algorithm to successfully solve the non-convex optimization function for the ensemble of models we propose.
- We demonstrate by a case study the validity of the model and an increase in the accuracy of employability prediction over extant methods.
- We show how our models can be interpreted to provide more leverage to HR personnel when deciding on the quality of candidates they want to hire. Additionally, we also demonstrate how test-taking candidates can gain from our models through the nature of feedback that can be generated from the models.

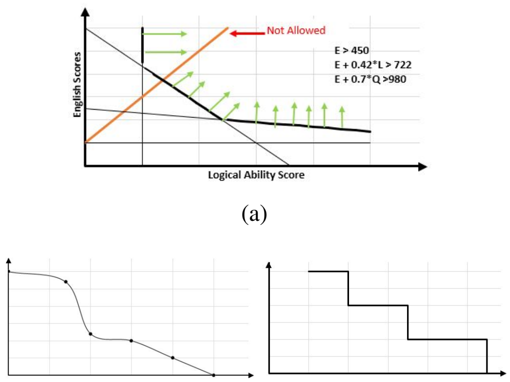
The paper is organized as follows: §2 describes the structure of the ensemble of classifiers we have designed. It also discusses Particle Swarm Optimization, the technique we use to solve the model's non-convex constraints. §3 lays out the objectives of our experiments. In §4, we present a case study of a real-world implementation and discuss our findings. Finally, §5 concludes the paper.

2. Classification model

2.1. Structure of the model

For our task, we need models which are coordinate-wise monotonic (CM), simple and human interpretable. A CM model has a monotonic relationship between each variable and the output, if one holds the other variables to any set of constant values. We propose to use a combination of sparse linear models with positive coefficients for the classification task. Stated another way, the model is a set of linear inequalities with positive coefficients, all to be simultaneously satisfied to give an output of 1. By sparsity we imply a large proportion of zero coefficients in every inequality. This is demonstrated for two variables (English and Logical ability scores) in Fig 1.a. The red line in Fig 1.a. represents a model with negative coefficients. Such models are not allowed in our system as they interpret to

[1] CM doesn't mean linearity. CM models can be non-linear and powerful.



(a)

(b) Non-linear curves (c) Decision tree

Figure 1. Coordinate-wise monotonic curves

candidates with lower scores being accepted over candidates with higher scores. For e.g. in Fig. 1, for a given Logical ability score, the model with negative coefficients rejects candidates with higher English scores while accepting the low scoring ones. One may observe that though the models are non-linear, they are simple, monotonic and interpretable. Figures 1.b. and 1.c. show other possible CM models.

Our models follow the theory of employee selection and matches how IOP experts have traditionally come up with selection criteria. They would put individual cut-offs on skills based on the minimum level of skill needed for a job. Then they would identify combinations of *compensatory* skills (Kleiman, 2014) where the lack of one skill can be compensated by another to demonstrate job suitability for a role. For e.g. the lack of extraversion in a person being considered for a sales and marketing profile is compensated by conscientiousness to be successful in the role. They would find multiple such compensatory relationships. In the above example, while one compensatory relationship is to do with personality traits, there could be another to do with language and cognitive skills.

In this paper, we demonstrate the use of this model for a two class classification problem - to predict whether a candidate would perform satisfactorily or not. Our technique is easily extensible to multiple classes. It is also extensible to related model classes such as an OR of the conditions rather than an AND.

2.2. Optimization criteria

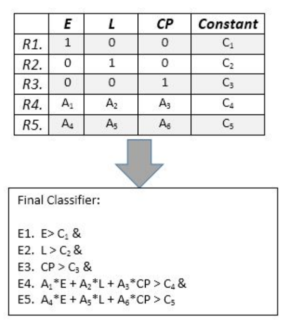
The optimization criteria used for our models were -

- **Minimize** $Type$**-1 error:** Type-1 error here is the ratio of the number of unsatisfactory candidates (referred to as UNSAT henceforth) classified as satisfactory to the predicted number of SAT.
- **Minimize** $Type$**-2 error:** Type-2 error here is the ratio of the number of SAT classified as unsatisfactory to the total number of SAT.

2.3. Particle swarm optimization

We wanted sparse and multiple models that traded type-1 and type-2. We cast the classification problem as an optimization problem to minimize the weighted sum of type-1 and type-2 errors. To do so, we use the classical form of Particle Swarm Optimization (PSO) to solve this problem[2]. PSO refers to a class of population based evolutionary algorithms (Parsopoulos & Vrahatis, 2002) which is quite effective in doing continuous parameter optimization. PSO begins with a swarm of particles as the initial population. Each particle has a position and a velocity. The position of the particle encodes the solution of the problem. The velocity of the particle represents the value added to the position of the particle to find its position in the next generation. The algorithm updates the position and velocity of all particles in each generation until the algorithm finds an optimum. The velocity of all particles is initially zero and is updated according to the best local position (best fitness) the particle has come across in its lifetime (all generations so far) and the best position any particle in the whole swarm has ever come across.

Figure 2 shows a sample structure of the particles representing an ensemble of linear predictors.



	E	L	CP	Constant
R1.	1	0	0	C_1
R2.	0	1	0	C_2
R3.	0	0	1	C_3
R4.	A_1	A_2	A_3	C_4
R5.	A_4	A_5	A_6	C_5

Final Classifier:

E1. $E > C_1$ &
E2. $L > C_2$ &
E3. $CP > C_3$ &
E4. $A_1{*}E + A_2{*}L + A_3{*}CP > C_4$ &
E5. $A_4{*}E + A_5{*}L + A_6{*}CP > C_5$

Figure 2. Particle structure

The coefficients A_is represent the weights on individual variables while the constants C_is refer to the thresholds. The weights are multiplied with the variables to form inequalities as shown in $E1$-$E5$, which are then optimized based on the optimization criteria. In rows 1-3, we keep the coefficients of one of the variables as 1 while the rest are set to 0, resulting in equations $E1$, $E2$ and $E3$. In rows 4-5, we learn the coefficients on all the variables as well as the corresponding constant, resulting in $E4$ and $E5$. These

[2]Any of the multi-objective continuous valued optimization heuristics may be used. We demo one technique.

form linear predictors with partial weights on each variable. We then take a logical AND of each of these predictors to obtain the final classification model. Considering only one equation in our ensemble of the type $E4$ or $E5$ results in a special case of the system mimicking logistic regression.

The fitness function used was of the form

$$Fitness(particle) = minimize(a * type1 + b * type2) \tag{1}$$

The weights of the multiple objectives were identified using dynamic weighted aggregation algorithm (Parsopoulos & Vrahatis, 2002). Those solutions which had a lower sensitivity i.e. which were at a larger distance from both classes were preferred. Some modifications were included in the algorithm to ensure that the solutions always have the exact form as described above. The coefficients of the solutions were probabilistically set to 0 according to the magnitude of the coefficient. This ensured sparsity of the solutions. Also, a backward elimination algorithm pruned the solutions which did not perform well. Here, the best trade-off points for type-1 and type-2 are considered by looking at the Pareto front (also known as the non-dominated front) of the type-1 vs. type-2 distribution generated by all the particles in the space. This means that for any given type-1 or type-2 respectively, the point on the Pareto front would have the least possible value for type-2 or type-1 respectively.

3. Experiments

We wish to answer the following questions with regard to our technique for building employability benchmarks -

- Could we build employability benchmarks with acceptable type-1 and type-2 errors using our techniques? If organizations use our benchmark for hiring, will they be able to reduce hiring unsatisfactory employees without adverse selection?

- Does an ensemble of linear models provide better prediction accuracy than a single, linear classification model?

- What insight and knowledge discovery may happen by studying our models? Could we discover what combination of parameters and variables determine employability for a job sector?

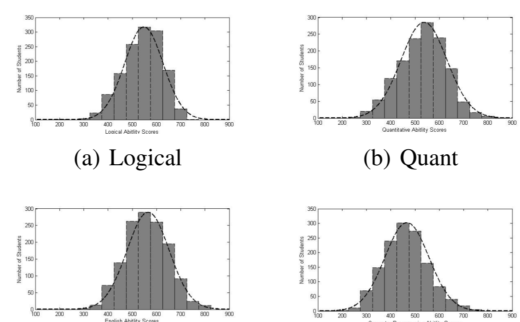
We explore the answers to these questions through a case study with a large IT services organization in India (referred to as ITCOM henceforth).

4. Case study

4.1. Problem statement

We used our ensemble of models to predict the performance of hires made by a large IT services company in India. Having over 100,000 employees, the ITCOM hires 10,000+ candidates annually. Candidates hired by the ITCOM first undergo an internal training program before being deployed on live projects. The training program lasts for three months and involves classes being conducted by training instructors on topics related to programming. During this program, they are administered a suite of assessments periodically to gauge their performance. At the end of the training program, only those candidates who meet certain threshold criteria in their internal assessments, termed *satisfactory* (SAT), are finally absorbed as full time employees. The rest, termed *unsatisfactory* (UNSAT), are considered to have failed the recruitment process and are rejected.

The company's objective was to hire *trainable* candidates and to be able to predict, at the time of hiring, candidates who would succeed in their three month training program. They aimed to minimize UNSAT candidates without actually eliminating SAT candidates. Predictive models which would optimize on this requirement would also help in building a benchmark for hiring in IT companies at large, providing them useful feedback to better understand the quality of their hires.



(a) Logical (b) Quant

(c) English (d) Programming

Figure 3. Candidate Distribution - AMCAT

4.2. Data set

In order to predict the training performance of candidates, a stratified sample of candidates was required to take AMCAT (AMCAT, 2015), an employability test designed by Aspiring Minds (Aspiring-Minds, 2015), as soon as they joined the company and before commencing their training program. AMCAT is a standardized, adaptive employability assessment which tests personality traits, language, cognitive and domain skills. It is taken by 100,000+ undergraduates and graduates every year in India, with more than 600 companies using these scores for shortlisting candidates for their recruitment processes. It's a computer based, proctored test which has different sub-tests for evaluating different skills. The tests are 30-35 minutes long and consist of approximately 25 questions each. Specifically, the ITCOM candidates were administered tests to measure the following skills - quantitative ability, logical ability, English comprehension skills and computer programming ability.

We collected scores on these four skills on 1371 candidates. Of these 1371, 946 (69%) were satisfactory candidates (SAT) whereas 425 (31%) were UNSAT. The input to our models were assessment scores of the candidates(see §4.3 for details). In order to build our models, we had 960 (70%) data points in the train set and 410 (30%) data points in the test set. The output had two variables, 1 or 0: 1 implying SAT and 0 implying UNSAT. We ensured a stratified sample in the train and the test set.

4.3. Observations

All results reported in the following sub-sections are on the test set.

Employability benchmark

We were interested to build a benchmark for employability for the software engineer profile in the IT services industry. We were looking for a model with a reasonable type-1 and type-2 error (<20%), which could also help the ITCOM hire better. When using all variables as an input to the model, we find a model with a type-1 error of 0.15 and type-2 error of 0.20 on the test set among our trade-off models. This fulfils our stated criteria. If the model is applied on the current test set, the percent of SAT candidates is 82% as compared to 69%. This indicates that if the company hires using this model, they shall be able to reduce UNSAT candidates by 40-50%. This clearly showed that the assessment scores could indeed predict training success and provide an employability benchmark.

Single vs. Multiple Linear Predictors

We first try to analyze whether an ensemble of linear predictors performs better than a single linear predictor. In Figure 4, we plot the type-1 vs. type-2 tradeoff curve for predictors learnt on all scores. The red line denotes the Pareto front (see §2 for details) obtained from a single linear predictor model while the blue line represents that obtained from multiple linear predictors. A linear predictor was obtained by considering only one equation with non-zero coefficients in our PSO tool[3] (see §2.3 for details). From the graph, one can clearly observe the additional accuracy achieved by using an ensemble model. The gap between the two fronts is more visible when one looks at a type-2 range of 15-30%. An incremental improvement of 4% on type-1 for a given type-2 would translate to avoiding 400 bad hires if 10,000 candidates are hired. This is indicative of the fact that an ensemble of predictors better models employability as opposed to a single linear model and provides better hiring accuracy to the ITCOM.

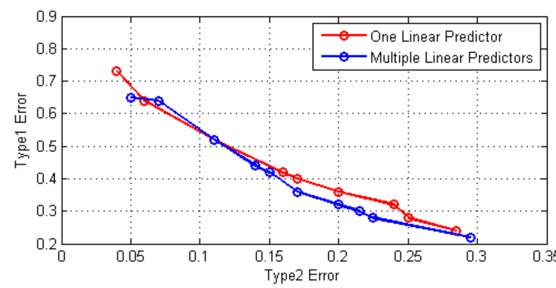


Figure 4. Single vs. Multiple Linear Predictors

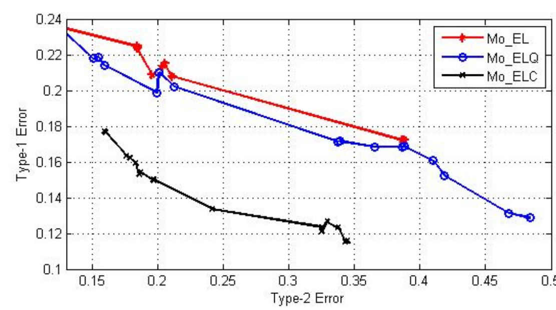


Figure 5. Type-1 vs. Type-2 between different models

What parameters are important to define employability?

In order to determine which variable is important and contributes to the prediction of post training success, we analyze the type-1, type-2 tradeoff curves of models built from different sets of variables. Table 1 describes the models built using different combination of variables.[4] We no-

Table 1. Variables used in different models

Model Name	Variables
Mo_EL	English, Logical
Mo_ELQ	English, Logical, Quant
Mo_ELC	English, Logical, Computer Programming

[3]We also performed logistic regression and picked the non dominating points among regression results and that of PSO

[4]We omit the model built using English and Quantitative ability scores. It was fully dominated by *Mo_EL*

tice that the quantitative ability scores add little significant value over logical ability scores towards prediction accuracy. On the other hand, a model with just English and quantitative ability performs worse than that with English and logical ability. This shows that once a candidate has his/her logical ability in a particular range, his success in the training is not dependent on his/her quantitative ability scores. It is sufficient to test a candidate on logical ability and companies do not really need a test on quantitative ability as a filtering parameter. It is interesting that a large number of companies do test quantitative ability other than logical ability and could possibly be experiencing type-2 error depending on the model they use. On the other hand, it would also be disenfranchising learners capable for the job out of the market.

We also see that Mo_ELC (marked in black) clearly outperforms all other models on both type-1 and type-2 errors. This means that the computer programming skills of a candidate at the time of entering the training program strongly predicts his/her success in the program. This observation is in variance with the current practice, where a majority of the top 10 IT companies in India do not use a programming test in their selection process. The general belief is that English comprehension and logical ability measures are sufficient to predict trainability and a cut-off on programming ability shall only increase type-2 error. Clearly, we find a different result and find that using a programming test will not only increase the number of SAT candidates, but also decrease the number of UNSAT candidates. In the next section, we look at the structure of actual models to find out how the test scores influence employability and investigate why the programming test scores help in defining employability better.

Table 2. Sample features and their interpretation

Model	Equation	Type-2 Error	Type-1 Error
PSO(E,L)	$L > 494$, $0.42 * E + L > 722$	0.21	0.21
PSO(E,L,Q)	$E > 405$, $L > 401$, $0.75 * Q + L > 874$	0.21	0.20
PSO(E,L,CP)	$CP > 360$, $0.47 * E + 0.98 * CP + L > 1196$	0.20	0.15

E : English comprehension, Q : Quantitative ability, L: Logical ability, CP : Computer programming ability

Model insight and candidate feedback Table 2 refers to specific equations from different Pareto fronts and their corresponding errors. Considering the $PSO(E, L)$ equation, one observes that the candidate needs to have a minimum level of logical ability to be trainable. Beyond this threshold, a lack of logical skills can be substituted by a high English score and vice versa. We also find that logical ability has a higher weight in the equation, which implies that to compensate every 10 points of the logical ability score, one approximately needs 20 extra points in the English score.

The $PSO(E, L, CP)$ model suggests that the candidate needs a minimum score in the computer programming test. A score of 360 is around a 30 percentile point on the national engineering norm and signals a basic level of familiarity with programming terms and concepts, according to the module rubric. Past this threshold, good logical and English skills are able to substitute for a lack of computer programming skills.

This implies that students need to have exposure and familiarity with computer programming to succeed in a short duration training in programming. If they have this basic exposure, their understanding of programming skills can be compensated by logical ability and English skills. On the other hand, if they do not have basic programming skills, then possessing other skills do not help. Given that the $PSO(E, L)$ model doesn't have the programming score parameter, it puts a much higher cut-off on the logical ability score to achieve the same type-1 error, thus increasing type-2 error.

The model structure also helps provide objective feedback to learners. For instance, it can predict if a candidate lacks any specific domain or cognitive skills and by how much. This can go a long way in providing guidance to students towards higher employability.

5. Conclusion

There is a strong need to objectively quantify employability for different job profiles, provide employability feedback to learners and facilitate an easy way to connect meritorious learning with matching job opportunities.Use of machine learning to predict the employability of a candidate has been attempted multiple times in the past and with considerable success. However the efficacy and scalablity of these methods have been limited due to lack of human intuitive models and methods to learn them.

We propose a class of simple, human interpretable, coordinate wise monotonic models to predict employability based on standardized assessment scores. We learn these models using particle swarm optimization technique and provide multiple trade off solutions. We showcase the efficacy of this method by using this model to improve the organizational structure of a large IT services firm in India. We not only are able to significantly improve the incumbent organizational structure but also are able to provide important insights into what constitutes employability for a software engineering profile. Given the interpretability of the models, it facilitates objective educational feedback to learners regarding their skill gaps. In future, we wish to apply our technique for job profiles in multiple companies and regions.

Acknowledgment

The authors would like to thank Bhimender Arora for his help in this work and for his invaluable suggestions.

References

AMCAT. Aspiring minds computer adaptive test (amcat), 2015. URL https://www.myamcat.com/about-amcat.

Aspiring-Minds. Which engineer gets a job, 2013.

Aspiring-Minds. Aspiring minds, 2015. URL www.aspiringminds.com.

Barrick, Murray R and Mount, Michael K. The big five personality dimensions and job performance: a meta-analysis. *Personnel psychology*, 44(1):1–26, 1991.

Cascio, Wayne F, Alexander, Ralph A, and Barrett, Gerald V. Setting cutoff scores: Legal, psychometric, and professional issues and guidelines. *Personnel Psychology*, 41(1):1–24, 1988.

Chang, Hsin-Yun and Xi, Lifent. Employee turnover: a novel prediction solution with effective feature selection. In *WSEAS International Conference. Proceedings. Mathematics and Computers in Science and Engineering*, number 3. Citeseer, 2009.

Economist. Got skills?, 2014.

Hunter, John E. Cognitive ability, cognitive aptitudes, job knowledge, and job performance. *Journal of vocational behavior*, 29(3):340–362, 1986.

Josh Bersin, Todd Tauber. Putting moocs to work. In *Putting MOOCs to Work: How Online Education Impacts Corporate Training*. Slideshare, 2013.

Kantrow, Alan. How business can help measure education outcomes that matter, 2014.

Kleiman, Lawrence S. Employee screening and selection, 2014.

Lan, Andrew S, Waters, Andrew E, Studer, Christoph, and Baraniuk, Richard G. Sparse factor analysis for learning and content analytics. *The Journal of Machine Learning Research*, 15(1):1959–2008, 2014.

Li, Yung-Ming, Lai, Cheng-Yang, and Kao, Chien-Pang. Incorporate personality trait with support vector machine to acquire quality matching of personnel recruitment. In *4th international conference on business and information*, pp. 1–11, 2008.

Lievens, Filip and Sackett, Paul R. The validity of interpersonal skills assessment via situational judgement tests for predicting academic success and job performance. *Journal of Applied Psychology*, 97(2):460, 2012.

Naveh, Yehuda, Richter, Yossi, Altshuler, Yaniv, Gresh, Donna L, and Connors, Daniel P. Workforce optimization: Identification and assignment of professional workers using constraint programming. *IBM Journal of Research and Development*, 51(3.4):263–279, 2007.

NER. National employability report - engineers annual report 2014, 2014.

Parsopoulos, Konstantinos E and Vrahatis, Michael N. Particle swarm optimization method in multi-objective problems. pp. 603–607, 2002.

Schmidt, Frank L and Hunter, John E. Development of a causal model of processes determining job performance. *Current Directions in Psychological Science*, pp. 89–92, 1992.

Sternberg, Robert J and Wagner, Richard K. The g-ocentric view of intelligence and job performance is wrong. *Current directions in psychological science*, pp. 1–5, 1993.

Terviö, Marko. Superstars and mediocrities: Market failure in the discovery of talent. *The Review of Economic Studies*, 76(2):829–850, 2009.