TUTA/IOE/PCU

Journal of the Institute of Engineering
January 2019, Vol. ***15 (No. 1):*** *1-13*
© TUTA/IOE/PCU, Printed in Nepal

Job-Candidate Matching using ESCO Ontology

Aman Shakya, Subhash Paudel
Department of Electronics and Computer Engineering, Pulchowk Campus
Institute of Engineering, Tribhuvan University, Nepal
Corresponding author: *aman.shakya@ioe.edu.np*

Received: Nov 2, 2018 *Revised: Dec 8, 2018* *Accepted: Dec 12, 2018*

Abstract: Skills management is one of the key factors to address the increasing competitiveness among different companies. Suitable knowledge representation and approach for matching skills and competences in job vacancies and candidate profiles can support human resources management automation through suitable matching and ranking services. This paper presents an approach for matchmaking between skills demand and supply through skill profiles enrichment and matching supply and demand profiles over multiple criteria. This work builds upon methods for profile modeling, information enrichment and multi-criteria matching. The main contribution of this work is a methodology for harmonization and enrichment of heterogeneous profile models and skill set description by making use of the standard ESCO ontology. Secondly, an algorithm is proposed for similarity matching across multi-criteria for discovering set of profiles that best fits the job description criteria. A prototype web-based system has been developed to implement the proposed approach and deployed online. The system has been tested with real IT jobs related dataset and validated against relevance scores provided by human experts. Experimental results show consistent correspondence between the similarity ranking scores produced by the system and scores provided by the human users.

Keywords: Job matching, Skills Ontology, ESCO, HR management, profile similarity

1. Introduction

Human resource managers rate the internet as an important recruitment channel and over half of all personnel recruitment is the result of on-line job postings. Although job portals are an increasingly important source for job applicants and recruitment managers, they still exhibit shortcomings in retrieval and precision as the job offers are in several syntactic formats, i.e. searches are subject to the ambiguities of natural language in job descriptions and lack relations to similar or interdependent concepts. Particularly, queries which are over-specified or inconsistent return no matches while relevant job offers could still be found if the consistency or specificity problem were to be resolved. If exact matches are lacking, worse alternatives must be often accepted or the original requirements have to be compromised [3]. The job provider has certain requirements when it comes to the skill set, experience, education and expected salary of the people they are hiring. The goal is to optimize one or more criteria in order to achieve the desired result. In human resource scenario there is a need of optimizing several criteria simultaneously [1].

This paper is focused on modeling a solution for information enrichment by utilizing human resource ontology ESCO and development of a matching approach for comparisons between applicant profiles and job openings (as shown in examples in Tables 1 and 2 below) with focus on skills, occupations, and experience. In determining relative suitability of applicants with different skill sets with regards to a specific job offer, number of research questions arises as:

- How does one select people from the database with the sought after skills?
- If no exact match is found, can a selection be done where people with similar skills to the sought ones can be recommended?
- How can CV be ranked with regards to their skills, and as such, compared to each other with regards to suitability to given job offer?

Table 1: Example of job offer criteria

Job Title	Skill Required	Experience	Qualification
Senior web programmer in .Net	Asp.Net, C#	>5	Bachelor in computer engineering
Ruby on Rails Developer	Ruby on Rails, MongoDB	>2	Bachelor in computer engineering

Table 2: Example of candidate profile

Name	Skills	Designation	Experience
Paolo	Web Programming, C, SQL, MVC, jQuery, Python, Java	Analyst Programmer	11
John	Architect, Technical Lead, Development Lead, .Net, Asp.Net, C#, VB, Jquery, MVC	Senior Programmer	6

The main objectives of this work are – a) to facilitate automatic matching and ranking of CVs against job vacancy descriptions based on relevance, b) to incorporate heterogeneous descriptions of similar skills in CVs and job descriptions, and c) to enable non-exact matching of CVs and job descriptions based on multi-criteria similarity.

To meet these objectives, the key contributions provided by this paper are as follows:

- Enriching and harmonizing skill sets in candidate profiles using the ESCO ontology
- An algorithm for matching CVs and job descriptions based on multiple criteria.
- Ranking of CVs based on relevance to the job requirements calculated using a multi-criteria similarity measure.
- A web-based tool for maintaining database of CVs and jobs and matching them.

2. Literature Review

Some state-of-the-art works in the field of recruitment by matching candidate skills and competencies with job description using ontological techniques, similarity matching and logic based techniques are discussed below. In [3], the authors present a query relaxation technique which is able to return results even in cases of inconsistent or overly specific queries which would return no results if run strictly. Sub-symbolic methods estimate the similarity between job and applicant descriptions. Symbolic approaches allow a more intuitive way to formulate and handle preferences and domain knowledge. But due to their partial preference order they cannot rank all results in practice like sub-symbolic approaches. The proposed query relaxation approach combines both these methods. This paper also demonstrates that by having data based on formal ontologies, one can improve retrieval.

Bird Mating Optimization method for one-to-n skill matching is proposed in [1]. The method finds the optimal combination of skills from two or more CVs that best satisfy a job description. In this approach the CV sets as well as the job description are described semantically by using a skill taxonomy. To evaluate the quality of a solution (i.e. a set of CVs that satisfies the job description considered) a fitness function is defined that evaluates the degree of semantic matching of the combination of skills part of the considered solution to the set of skills of the job description. In [2], the authors propose an ontology based hybrid approach that matches a job seeker to job offers where the job seeker and job offers are described semantically by using a skill ontology, and the type of match is determined by using a description logic based classification. Additionally, a similarity based strategy is used in order to compute the degree of similarity between each of the job seekers and job description to rank the seekers according to their similarity scores.

The authors in [5] explain logic-based techniques and technologies that make recruitment process more efficient and flexible. And the system they propose automatically performs matchmaking between available candidate profiles and vacant job positions according to mandatory requirements and preferences provided by a recruiter. In order to perform it, we need a language suitable for data intensive applications with a good compromise between expressiveness and computational complexity. The system performs non-exact match through top-k retrieval technique: it uses a match engine which performs top-k queries over a DLR-lite Knowledge Base providing a ranked list of candidates. A hybrid Ant Colony Optimization based method for solving multi-skill resource-constrained project scheduling problem is proposed in [4]. In this approach, every edge in a path (i.e. solution) that the ant tries to build is represented as a given task together with the resources that are capable of performing the task, the pheromone value is a value that specifies the probability of assigning a given resource to given task, the path (i.e. a solution) is a set of tasks and their associated resources, while the surface is represented by the set of all feasible solutions. There is a special ant that leaves much more pheromone than any other ant in the colony. This ant is selected by using a Tabu-Search strategy.

The paper [6] illustrates a method for matching jobs to workers, which is able to deal with incomplete and inaccurate information. This approach is based on a probabilistic weighted ontology model that assigns weights to different attributes (i.e. location, skills, and qualification) and is able to perform a probabilistic conversion of audio content to text. In the case of location as attribute, the Euclidean distance is used to compute the distance between two points, while in the case of skills as attributes, a WordNet based strategy is applied to establish the distance between two skills. In the case of qualification as attribute, a lattice based approach is used. The quality of the method proposed has been evaluated by using a set of metrics from information retrieval.

This paper proposes another approach for matching CVs and jobs based on skill and experience using the standard ESCO ontology along with metric similarity measure, whereas, most of the previous works is focused on development of standard ontology for human resource management and utilization of logical reasoning techniques for inferring new relationships from existing knowledge. The information enrichment methodology formulated provides results even in case of incomplete skill sets in both the job and CV profiles. Even though the metric similarity approach followed is not completely novel, implementation and experimental validation of such technique in HR management domain has not been much pursued.

3. The ESCO Ontology

ESCO[1] (European Skills, Competence and Occupation) is a multilingual classification of European Skills, Competences, Qualifications and Occupations. It identifies and categorizes skills, competences, qualifications and occupations relevant for the EU labor market and education and training, in 25 European languages. The system provides occupational profiles showing the relationships between occupations, skills, competences and qualifications. ESCO has been developed in an open IT format, is available for use free of charge by everyone and can be accessed through an online portal.

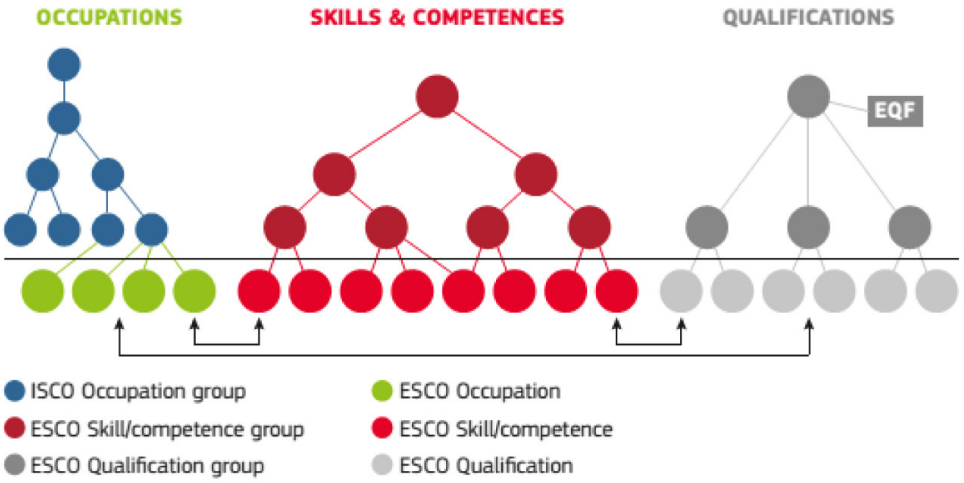


Fig. 1: ESCO data model

[Source: https://ec.europa.eu/esco/portal/escopedia/ESCO_data_model]

ESCO is structured on the basis of three interlinked pillars representing a searchable database in 25 languages. These pillars are – a) Occupations, b) Skills and competences, and c) Qualifications (certifications) as shown in the ESCO data model Fig. 1 above. Occupational profiles show whether skills and competences are essential or optional and what qualifications are relevant for each ESCO Occupation. The skills pillar contains knowledge, skills and competences as well as some group concepts. In ESCO v1 it contains about 13,500 concepts and is not organized in a full hierarchy, but structured through its link to occupations. The qualifications pillar allows Member States and awarding bodies to provide data on qualifications which is collected in ESCO. The qualifications are structured using the European Qualifications Framework (EQF) and the ISCED Fields of Education and Training 2013.

4. Methodology

An overview of the methodology is presented in Fig. 2.
The main steps of the methodology are as follows:

1. The first step in methodology is data collection which includes data set for both the job

[1] *https://ec.europa.eu/esco/portal/escopedia/Main_Page*

seekers and recruiters. The base data type for the job seekers is collection of CVs while for the recruiters is job description and necessary requirements.

2. In order to harmonize data collected from different sources, the next step is cleaning, integration, selection and transformation to convert heterogeneous data into one uniform data model. The data after pre-processing contains the personal skill sets, experience, education and expected salary data.
3. These pre-processed data of CVs and Jobs can be semantically linked by using domain ontology of skill set (ESCO). The main purpose of this semantic linking is to find semantic distance between different terms that are used for defining skill set.
4. The next step is generation of suitable matching CVs in correspondence to the job description. This is done by filtering out widely dissimilar CVs found by computing metric distance between skill sets in the CV and job description.
5. Next step is finding best fit between the CV and job description by considering multiple criteria that define the job description. The CVs are ranked based on the fitness score and results are provided.

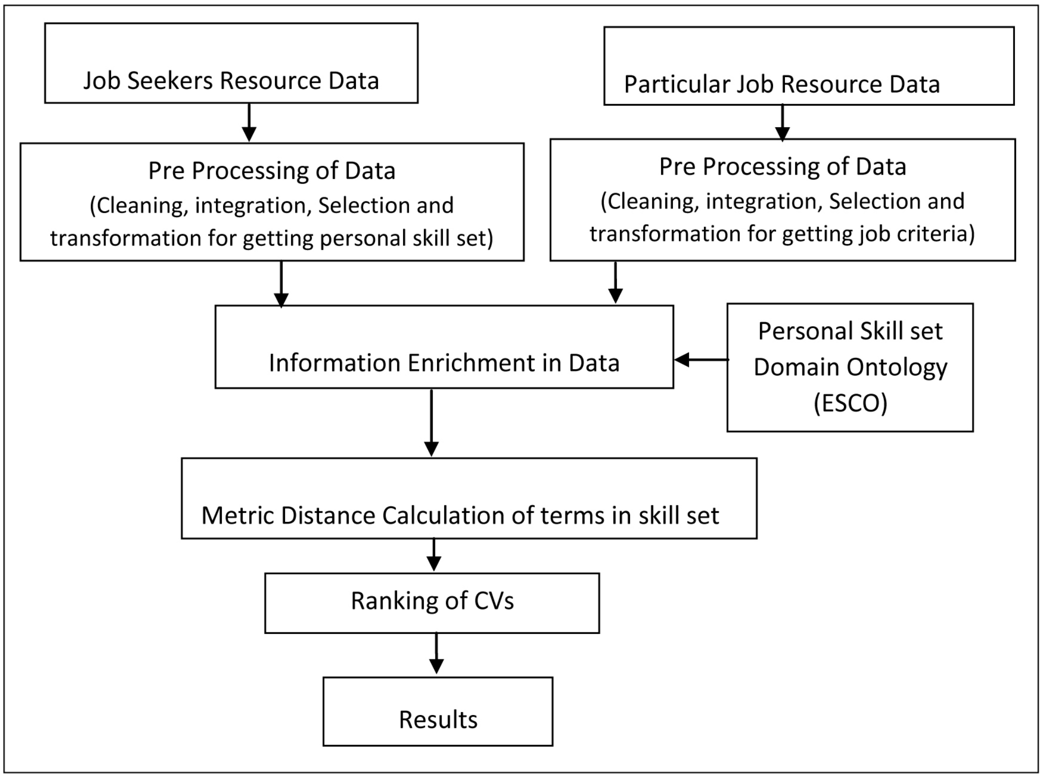

Job Seekers Resource Data
Particular Job Resource Data
Pre Processing of Data
(Cleaning, integration, Selection and transformation for getting personal skill set)
Pre Processing of Data
(Cleaning, integration, Selection and transformation for getting job criteria)
Information Enrichment in Data
Personal Skill set Domain Ontology (ESCO)
Metric Distance Calculation of terms in skill set
Ranking of CVs
Results

Fig. 2: Overview of the methodology

The major parts of the methodology are described in detail below.

4.1 Model Development

The basic models for representing different entities that will be part of the match making process are:

Job seekers: Job seekers represent the group of people who are looking for jobs in the domain of their skills. The number of job seekers in the market is . Each job seeker owns a set of non-transferable endowments: they cannot be exchanged among job seekers. The set of endowments for individual is termed as CV and is characterized by skill vector and experience. Skill vector for individual i is characterized by , where K is the number of skills.

Experience for individual *i* is characterized by *ei*= Skills are heterogeneously distributed among the job seekers, so the vectors are a multivariate random variable and represents the set of skills available in the specified domain.

Employers (Companies or Firms): This represents the companies that have various businesses in different domains. The number of firms in the given business domain is represented as and have different hiring requirements which is represented as job description.

Job Description: This represents way a firm opens vacancies for different positions that are available at the firm. The job description is represented as skill vector exactly as used for the job seeker.

4.2 Metric Similarity Model

Similarity score is the measure to show how similar two set of data are to each other. For similarity measure the method of rewarding common substrings and a common ordering of those substrings have been used. The proposed algorithm considers not only the single longest common substring, but other common substrings too. The algorithm answers the question “Find out how many adjacent character pairs are contained in both strings". By considering adjacent characters, the algorithm takes into account not only the characters, but also of the character ordering in the original string, since each character pair contains a little information about the original ordering. This metric similarity measures similarity over strings by splitting them up into their character pairs and using the following relation.

4.3 Mapping to ESCO Model

For a given skill set si in Job/CV the skill set is expanded by mapping with skills defined in the ESCO ontologies by considering the similarity between the skill s(j, i) and s(m, esco) where s(j, i) is a skill in CV and s(m, esco) is the skill from the ESCO ontology. If w(j, m) is the similarity between the skills then the skill is added in the skill set si by adding the weight to the skill in the skill set vector. In the next iteration similar mapping is done with the similar skills that are related to the occupations that correspond to the skills in the skill set si. After this mapping the skill set vector is enriched with necessary similarity weight, which is represented as:

si = (s(1,i),w(1,i)),..,(s(M,i),w(M,i))
s. t. i.e. the original count of skills in the skill set from the CV.

This expansion of skill set will give us more detailed association with various skill sets that have been defined for different domains of works from the ESCO ontology, thus helping to build model that can lead towards better match between skill sets in demand (job descriptions) and offer (CVs)

4.4 Mapping of Jobs and CVs

Matching between skills of individuals and job description is defined by a number of factors.

Negative Information treatment: This factor affects the choice of the language in which descriptions have to be expressed and is fundamental in the matching process of any kind of description. The possibilities can be itemized as follows:

Absent: all information allowed in profile descriptions are positive and all others are considered unknown.

Implicit: lacking information in a description is implicitly managed as negative.

Explicit: negative information can be elicited in descriptions together with positive ones, but all not elicited information are considered unknown.

Multiplicity of Relationship between Individuals and Jobs: In case of skill matching, one offered profile may be assigned more than one task and vice versa. Match relationship between Individuals and Jobs may be characterized by one of the following multiplicity:

One to one: There is a one job profile to match with one individual; offered and requested profile descriptions may be relative to more than one skill.

Many to one: There is a one job to assign to several people. This happens for example in the selection of a working team for a project, representing in this case the task to assign.

One to many: If there is a search for one individual attending to many simple tasks. The scenario is similar to time-sharing in Operating Systems, in which one resource need to be shared between several users. In this context many tasks share the same human resource.

Many to many: If there are many tasks to assign and many individuals available, the search is for the best scheduling of human resources on the different tasks.

In this work the considered scenario is only limited to Negative Information treatment – absent and Multiplicity of Relationship between Individuals and Jobs – one to one. The Negative Information treatment – explicit is partially handled during the skills expansion by utilizing the ESCO taxonomy as explained in the sub-section mapping to ESCO ontology.

4.5 Algorithms

The two main algorithms used in our approach are listed below.

Algorithm 1: Enrich metadata of Jobs/CVs

```
Step 1: START
Step 2: Take Skill set (array of skill) of Jobs or CVs and ontology as input.
Step 3: Find the metric similarity between the skill from skill set and each skills from the ontology
Step 4: Case I - If the skill is from the ontology, make a skill object using the similarity weight,  URI of the skill from ontology and name of skill from the ontology.
Case II - If the skill is from the skill set, make a skill object using the similarity weight, URI of the skill from skill set and name of skill from the skill set.
Step 5: Repeat from Step 3 for each skill in skill set to get the array of skill object
Step 6: Choose the skill object array having similarity weight greater than ‘α'
Step 7: For each skill object find the corresponding occupation using ontology
Step 8: For each occupation from step 7 find corresponding skill object array as in step 5 using Ontology. Multiply the similarity weight by factor ‘$\beta$' for this skill object Array.
Step 9: Choose the skill object array having similarity weight greater than ‘$\alpha$'
Step 10: Merge the skill object array from step 6 and step 9 then final skill object array is generated as output
Step 11: STOP
```

Algorithm 2: Rank the CVs

```
Step 1: START
Step 2: Take CVs object array (arrCVObj), Job object array (arrJobObj), Job Input, Experience Criteria (EC) as input
Step 3: For Job Input find the corresponding job object (JobObj) from Job object array (arrJobObj)
Step 4: Take CV object (CVObj) from CVs Object array (arrCVObj)
IF CVObj.experience satisfy EC
    Using skillweight create new CV object CVObjRank
END IF
Step 5: Repeat step 4 for each CVObj in arrCVObj and create the array of CVObjRank
Step 6: Rank CVObjRank array according to skillweight and select no of ranked CV as needed.
Step 7: STOP
SumofSkillWeight=(Find Metric similarity of CVOnj.skill and JobObj.skill)/(1000*(1+CVObj.experience) )
SkillCompareWeight= Find Metric similarity of job.jobInput.skill and CV.CVInput.skill
skillWeigh= SumofSkillWeight+SkillCompareWeight*$\gamma$+CV.expereince*$\lambda$
```

4.6 Dataset and Validation

Jobs and CVs raw data for this work are taken from the job advertisement site *www.hirefire.co.uk* for the period of 5 years. These data are in the form of SQL tables so they can be easily processed. The CVs taken initially for the experiment are 1060 and the jobs taken are 110. New CVs and Jobs can be added to the system manually as well. The ontology used in this work is ESCO ontology which identifies and categorizes skills, competences, qualifications and occupations relevant for the EU labor market and education and training. 7 human evaluators were chosen among people who were IT professionals having experience of at least 5 years, to ensure that they were good at ranking CVs for IT job requirements. This allowed us to simulate the candidate selection as it happens in professional companies. The validation data given to the individual human evaluator included the criteria and job description and list of CVs. Each CV had an identifier, skill set and experience which were generated from the system.

5. System Implementation

The technical architecture of the developed system is shown in the above Fig. 3. The system has been developed mainly using C#, .Net Framework and JSON Parser. The front-end is developed using ASP .Net and backend uses MS MSQL server database.

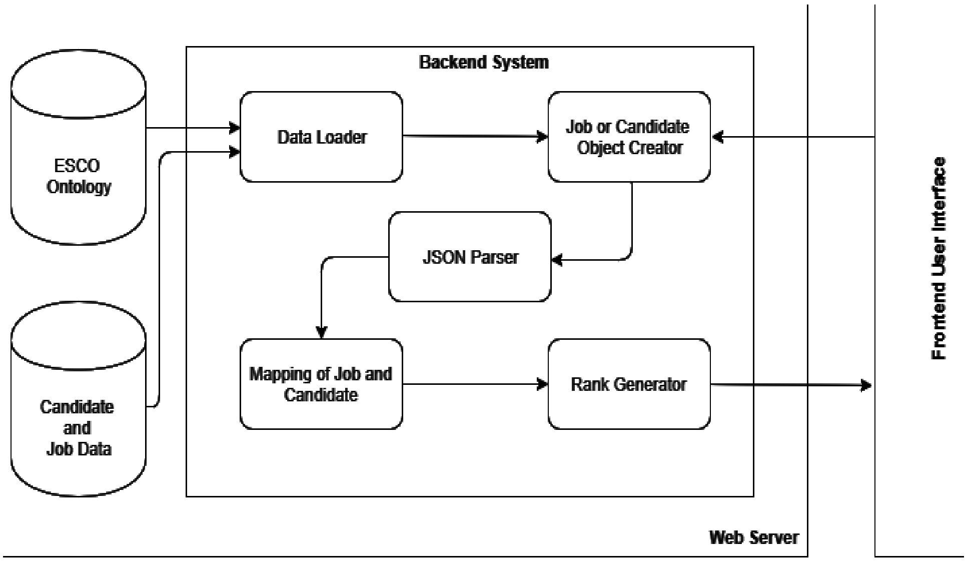
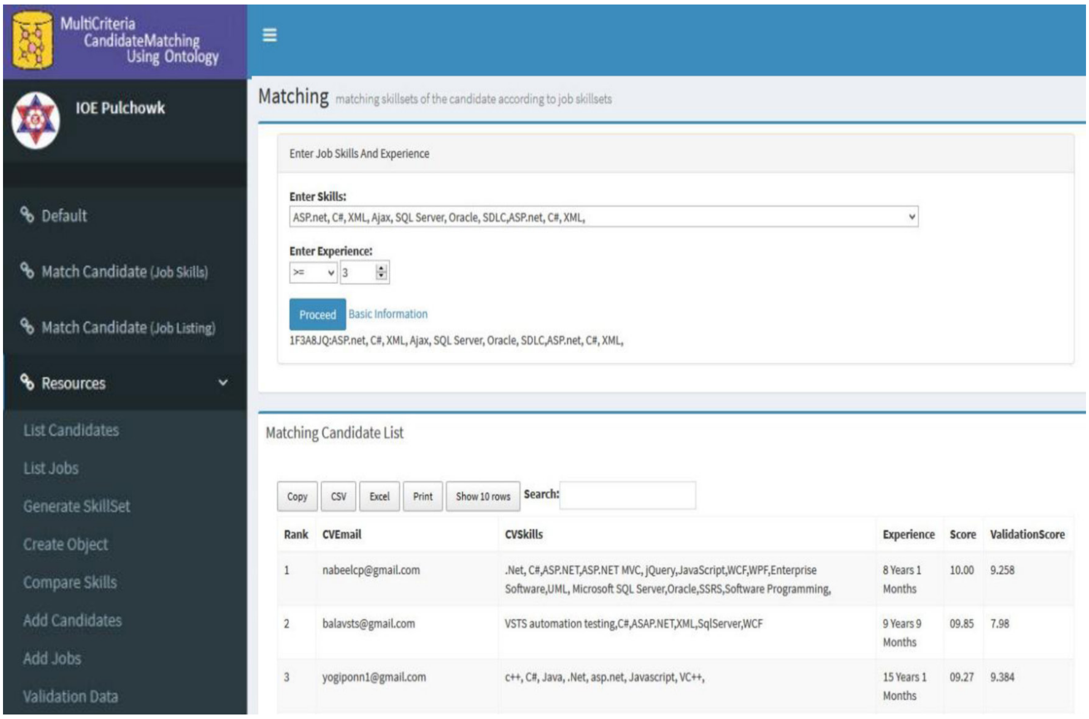


Fig. 3: System architecture

A web-based system implementing the proposed approach for job-candidate matching was developed and deployed online. The prototype implementation is available online at *http://202.70.67.149:8383*. It has been hosted in an online server at the ICT center at the Institute of Engineering, TU, Nepal. When the skill set of the job description as well as experience criteria is given to the system the outputs are the list of ranked CVs with similarity score matching the particular job. The Fig. 4 is the basic output for matching the candidate with similarity score to the given job description.



Matching matching skillsets of the candidate according to job skillsets

Enter Job Skills And Experience

Enter Skills:
ASP.net, C#, XML, Ajax, SQL Server, Oracle, SDLC,ASP.net, C#, XML,

Enter Experience:
>= 3

1F3A8JQ:ASP.net, C#, XML, Ajax, SQL Server, Oracle, SDLC,ASP.net, C#, XML,

Matching Candidate List

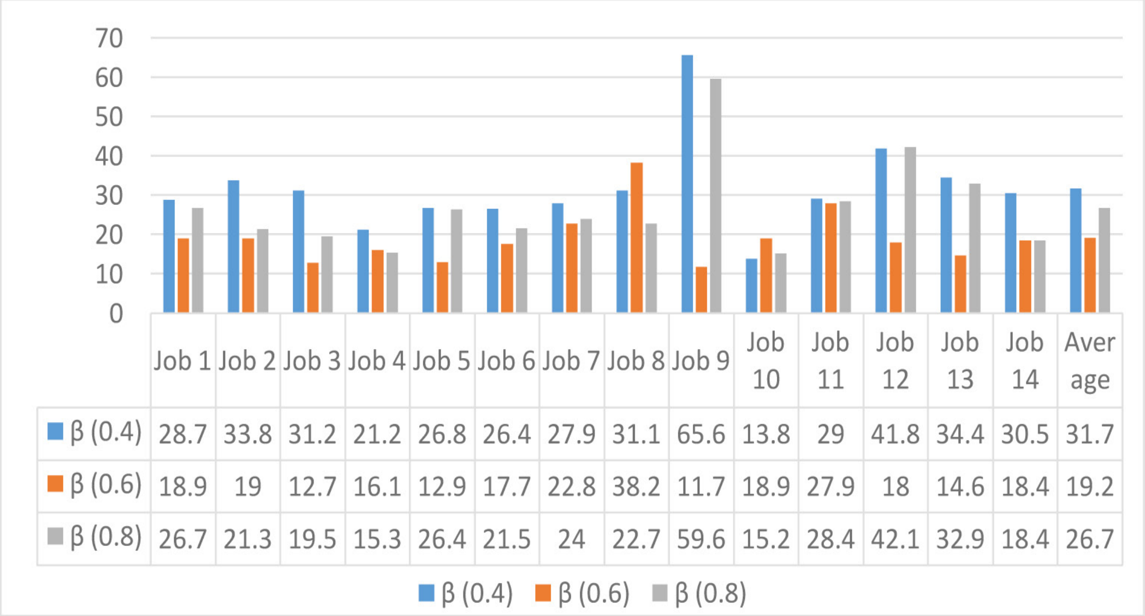
Search:

Rank	CVEmail	CVSkills	Experience	Score	ValidationScore
1	nabeelcp@gmail.com	.Net, C#,ASP.NET,ASP.NET MVC, jQuery,JavaScript,WCF,WPF,Enterprise Software,UML, Microsoft SQL Server,Oracle,SSRS,Software Programming,	8 Years 1 Months	10.00	9.258
2	balavsts@gmail.com	VSTS automation testing,C#,ASAP.NET,XML,SqlServer,WCF	9 Years 9 Months	09.85	7.98
3	yogiponn1@gmail.com	c++, C#, Java, .Net, asp.net, Javascript, VC++,	15 Years 1 Months	09.27	9.384

Fig. 4: Screenshot of the implemented online system

6. Experimental Results

For the purpose of validation and analysis, difference between the score given by system and evaluators' average score was calculated as absolute error. The Score_Actual provides the score for each CV calculated by the system, while the Score_Avg provides the average score given by all the evaluators.



	Job 1	Job 2	Job 3	Job 4	Job 5	Job 6	Job 7	Job 8	Job 9	Job 10	Job 11	Job 12	Job 13	Job 14	Average
🟦 β (0.4)	28.7	33.8	31.2	21.2	26.8	26.4	27.9	31.1	65.6	13.8	29	41.8	34.4	30.5	31.7
🟧 β (0.6)	18.9	19	12.7	16.1	12.9	17.7	22.8	38.2	11.7	18.9	27.9	18	14.6	18.4	19.2
⬜ β (0.8)	26.7	21.3	19.5	15.3	26.4	21.5	24	22.7	59.6	15.2	28.4	42.1	32.9	18.4	26.7

Fig. 5: Error comparison for different values of β

In the algorithms implemented, the coefficient α is threshold match score between skills in CVs or Jobs to ontology skills. The factor β is used to evaluate skill weight during enrichment of skills from the second layer ontology. The factor γ is used to evaluate skill which matches the skills of CV and Jobs without any enrichment. The factor λ is used to evaluate the impact of experience. The value of α is dependent on the number of skills considered. The higher the value of α lower the number of skills to be selected and the purpose is to filter out unrelated skills. From the experimentation value of α chosen was 0.3. The value γ gives the importance to the skills that are usually written in unstructured Job and CV skills. In order to use data coming from unstructured sources lower value for this coefficient give the good result and for the data obtained from standard CVs and Jobs higher value of γ give good results. For this work the value chosen as 2. The value of λ can be freely chosen based on importance of experience. This value was taken as 0.1 on the basis of different experiments.

From the experiment on changing the value of β it is observed that the data are more close to the expert score for β =0.6. So for this work the value chosen is 0.6. The following figures give the comparison of system and evaluators scores. The errors observed were approximately 18.94%, 12.71%, and 16.05% for three experiments respectively shown in figures below. It was noted that the CVs which have been ranked highest have smaller error to the CVs which have been ranked lowest. Also the profiles having different skill set described in the job also profited and ranked higher.

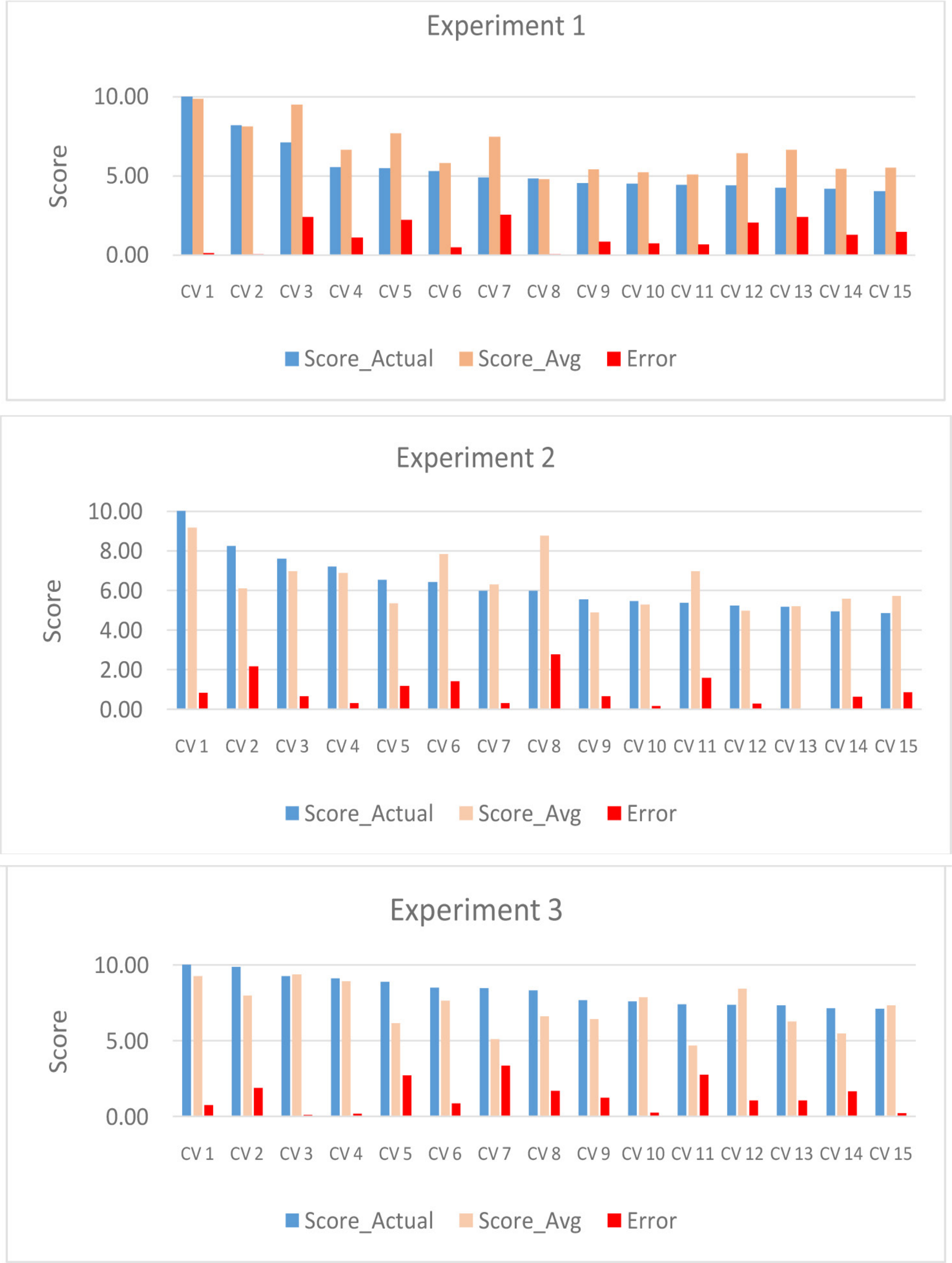


Fig. 6: The comparison of system and evaluator score for 3 rounds of experiments

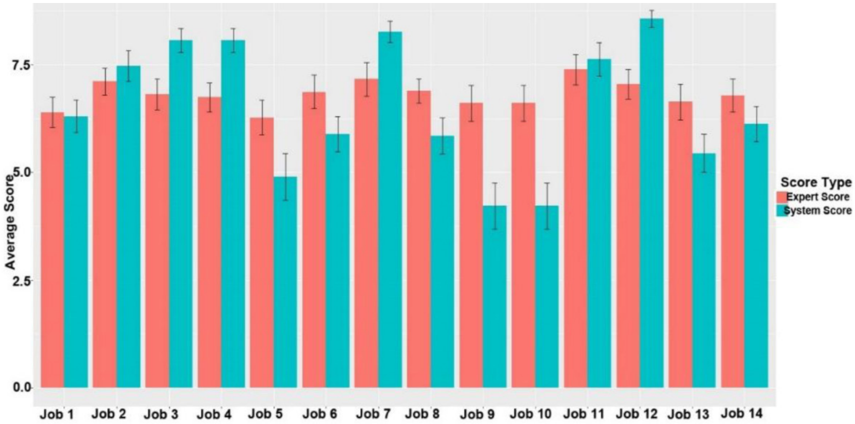


Fig. 7: The graph for the standard error

The graph below shows the standard error plot for both system and evaluator score. The results in the system show the even distribution of deviation of score from system and evaluator score across all job description which shows the system is consistent. Error in each job is listed in table 3 below.

Table 3: Overall error for each evaluated job

ID	Job 1	Job 2	Job 3	Job 4	Job 5	Job 6	Job 7	Job 8	Job 9	Job 10	Job 11	Job 12	Job 13	Job 14	Average
Error %	18.94	19.03	12.71	16.05	12.86	17.65	22.76	38.17	11.68	18.94	27.92	17.96	14.58	18.42	19.12

It can be observed that there is even distribution of error across job descriptions. Even though average error percent is 19.12% the performance can be improved by tuning factors in the matching algorithm.

7. Conclusion

In this paper, an approach is presented for matchmaking between skills demand and supply. The key contribution is enrichment of heterogeneous profile models and skill set description of candidate and jobs using ESCO ontology. An algorithm is proposed for similarity matching across multi-criteria to best fit job criteria. The result obtained was validated with scores from experts. Validation indicates quite good results with evenly distributed deviation.

Some recommendations for future are:

- The system can be improved by automatically linking with the evolving ESCO ontology.
- The matching algorithm can be further improved by including other criteria such as degree, previous projects, expectations, designation, soft skills, location etc.
- The system can also be extended with a learning algorithm to improve matches over time.

References

[1] Corde S, Chifu VR, Salomie I, Chifu ES and Iepure A (2016), Bird Mating Optimization method for one-to-N skill matching. In *Proc. - 2016 IEEE 12th Int. Conf. Intell. Comput. Commun. Process. ICCP 2016*: 155–162.

[2] Fazel-Zarandi M and Fox MS (2009), Semantic matchmaking for job recruitment: An ontology-based hybrid approach, In *CEUR Workshop Proc.*, **525:** 1–8.

[3] Mochol M, Wache H and Nixon L (2006), Improving the recruitment process through ontology-based querying, In *CEUR Workshop Proceedings*, 2006, **226**: 59–73.

[4] Myszkowski PB, Skowroński ME, Olech ŁP and Oślizło K (2015), Hybrid ant colony optimization in solving multi-skill resource-constrained project scheduling problem, *Soft Computing*, **19(12)**: 3599–3619.

[5] Straccia U, Tinelli E, Colucci S, Di Noia T and Di Sciascio E (2009), A system for retrieving top-k candidates to job positions, In *CEUR Workshop Proc.*, 477.

[6] White J, Kummamuru K and Rajput N (2012), Expansion Methods for Job-Candidate Matching Amidst Unreliable and Sparse Data, In *COLING'12 - Proc. 24th Int. Conf. Comput. Linguist.*, no. December: 1321–1330.