Bidirectional Job Matching through Unsupervised Feature Learning

DISSERTATION

zur Erlangung des Grades eines Doktors
der Naturwissenschaften

vorgelegt von

M.Sc. Sisay Adugna CHALA

eingereicht bei der Naturwissenschaftlich-Technischen Fakultät
der Universität Siegen
Siegen 2017

1. Gutachter: Prof. Dr.-Ing. Madjid Fathi
2. Gutachter: Prof. Dr. Kristof van Laerhoven
Vorsitzender: Prof. Dr. Volker Blanz

Tag der mündlichen Prüfung: 15.01.2018

gedruckt auf alterungsbeständigem holz- und säuerfreiem Papier

“Prediction is incompatible with choice in the case where you yourself predict what you will choose, or I predict and then tell you"

Ludwig Wittgenstein

Acknowledgements

My deepest gratitude goes to my parents who laid the foundation for me to get here. I would also like to thank my lovely better-half Aynalem Tesfaye, and my cute kids Edna and Mathias for their encouragement, support and tolerance.

I am grateful to Prof. Dr-Ing. Madjid Fathi, Chair of the Institute of Knowledge Based Systems and Knowledge Management (KBS and KM), Department of Electrical Engineering and Computer Science, University of Siegen under whose close supervision this research thesis was done to completion. I would also thank my second supervisor Prof. Kristof van Laerhoven, Chair of Institute for Ubiquitous Computing, Department of Electrical Engineering and Computer Science, University of Siegen who provided me with valuable advice and feedback on my dissertation. I would like to thank Prof. Kea Tijdens and all her team in Amsterdam Institute of Advanced Labour Studies, University of Amsterdam, for her enormous contribution in supervision, provision of data for this research and hosting me during my secondment phase.

My heartfelt thanks also go to all colleagues in the institute of KBS and KM who have been giving me ideas and encouragements during the course of this study. My special thanks go to Dr.-Ing. Fazel Ansari, Dr. Scott Harrison and Dr.-Ing. Mareike-Jessica Dornhöfer for their unreserved support in reviewing my work and providing me with valuable comments.

I would like to acknowledge the generous financial support of the Marie Curie ITN Project - "Crossing borders in the comprehensive investigation of labour market matching processes: An EU-wide, trans-disciplinary, multilevel and science-practice-bridging training network(EDUWORKS)" - (PITN-GA-2013-608311) which is part of the European Commission's 7th Framework Programme and all members of Eduworks for the lovely time we spent together and for their encouragements, sharing useful ideas and resources.

Last, but not least, I would like to thank all the Eduworks team for sharing ideas and experiences. My special thanks go to Dr. Stefan Mol and Dr. Gábor Kishmihok, Institute of Job Knowledge Research in the University of Amsterdam for dedicating their precious time to review my work whenever I requested them to.

Abstract

Job matching is a process that involves decision to whether a job vacancy is relevant, given the profile of job seeker and vice versa. It requires thorough understanding of job seeker and vacancy in order to match them bidirectionally. Bidirectional matching through measuring the degree of semantic similarity of job descriptions in vacancies and candidate job seekers has been a challenging task in the job recruitment industry. The challenges are associated with i) lack of information due to job seeker inability or resistance to provide sufficient data, ii) difficulty in modeling job seekers and/or vacancies and iii) the complexity of matching process itself.

Fortunately, the Internet and advancement of information technology provide opportunities that help deal with these challenges. Availability of huge online data about job descriptions which has been entered by job seekers and job holders can be utilized to understand job seekers. Large volume of online vacancy data can be exploited to portray the current demand of the job market. Prevalence of technological advancements to handle the size of big data and the complexities of the matching process makes job matching feasible to address.

Understanding the job seeker presupposes obtaining more information about the job seeker directly from himself, i.e., through resumé and web survey, or through others, i.e., from social network. This research investigates tools and techniques, and implements a web-based user interface, i.e., a context-aware Dynamic Text Field (DTF), that allows users to enter data of their choice but with guidance using autocompletion. Moreover, this study identifies methods that measure the skills, expertise and experience of a job seeker and investigates the importance of using social networking data as input to user modeling that determines the strength of skills to be used for recommending matching job vacancies.

In addition to job seekers, job matching requires understanding and modeling of vacancies. Though online vacancies are publicly available, due to overwhelming volume of data, job seekers are not able to easily find relevant vacancy for their skill or are unable to analyze the requirements to estimate its relevance. Analyzing vacancies as well as optimizing the matching process, on one hand, and exploiting the available opportunities of big data and technological advancement, on the other hand, are of paramount importance to pursue a novel approach of job matching.

This research employs solutions that learn from data (as opposed to rules) because they perform better at handling job seekers and vacancy data in the ever changing market. One of the methods to address these challenges is applying Machine Learning – data-intensive techniques to model job seekers and vacancies – and get a better matching. It explores matching job vacancies with job seekers using data

from online vacancies, occupational standards, resumés, job seeker's self-assessment, and social network data. Deep unsupervised feature learning, which is a kind of machine learning, is applied to develop a novel bidirectional matching of job seeker and vacancy through modeling the former using data from self-assessment, resumé parsing and social network, and the latter using vacancy parsing and enriching it via occupational standards. The choice of the data is based on its suitability to model different aspects of job seeker and vacancy. Machine learning is chosen because of the dynamics of job market, i.e, the jobs change so frequently that developing rules is practically infeasible, whereas learning from data is feasible with the availability of large online data, and robust computational resources.

The results of this endeavor are i) development of algorithm for context-aware DTF for user profile survey; ii) a new technique to measure skill relevance using social network-enhanced job seeker modeling; iii) improved relevance ranking of job vacancies through feature enriching by job titles and descriptions from standard occupations.

Zusammenfassung

Job-Matching ist ein Prozess der angesichts des Profils eines Arbeitsuchenden überprüft inwiefern eine Stellenausschreibung maßgeblich ist und umgekehrt. Bidirektionales Matching durch Messung des Grades der semantischen Ähnlichkeit von Berufsbeschreibungen in Stellenangeboten und Bewerberprofilen ist eine fortlaufende Herausforderung für öffentliche und private Arbeitsvermittlungseinrichtungen. Diese Herausforderungen sind i) der Mangel an Informationen aufgrund der Unfähigkeit oder dem Widerstand des Arbeitssuchenden ausreichende Daten zur Verfügung zu stellen, ii) Schwierigkeiten bei der Modellierung von Profilen von Arbeitsuchenden und/oder Stellenangeboten und iii) mit der Komplexität des Matching-Prozesses selbst assoziiert.

Glücklicherweise bieten das Internet und die Weiterentwicklung der Informationstechnologie Möglichkeiten mit diesen Herausforderungen umzugehen. Die Verfügbarkeit großer Online-Datenbanken über Arbeitsplatzbeschreibungen, die von Arbeitsplatzsuchenden und Arbeitnehmern eingegeben wurden, können zum besseren Verständnis von Arbeitssuchenden Anwendung finden. Ein großes Volumen an Online-Stellenangeboten kann zudem genutzt werden, um den aktuellen Bedarf des Arbeitsmarktes darzustellen. Die Prävalenz technologischer Fortschritte macht es möglich Big Data und die Komplexität des Matching-Prozesses zu adressieren und ein automatisiertes Job-Matching zu erarbeiten.

Das Verständnis über den Arbeitssuchenden setzt voraus, dass weitergehende Informationen direkt von oder über diesen ermittelt werden, zum Beispiel durch dessen Lebenslauf, eine Suche im Web, oder durch andere Kanäle wie etwa soziale Netzwerke. Die vorgelegte Arbeit untersucht Werkzeuge und Techniken zur Ermittlung und Erfassung weiterer Informationen und implementiert hierzu eine webbasierte Benutzeroberfläche in Form eines kontextbezogenen dynamischen Textfelds (DTF), in welches Benutzer Daten ihrer Wahl eintragen können und dabei durch Funktionen zur Autovervollständigung unterstützt werden. Außerdem identifiziert die vorgelegte Dissertation Methoden die die Fähigkeiten, Fachkenntnisse und Erfahrungen eines Arbeitssuchenden messen. Des Weiteren wird die Bedeutung der Verwendung von Daten aus Sozialen Netzwerken als Input für die Benutzer-Modellierung untersucht, um davon abhängig die Stärke der Fähigkeiten zu bestimmen und diese letztendlich für die Empfehlung passender Stellenangebote zu verwenden.

Zusätzlich zu dem Verständnis über den Arbeitsuchenden, erfordert der Job-Matching Prozess das Verständnis von Stellenangeboten und deren Strukturierung bzw. Modellierung. Obwohl Online-Stellenangebote öffentlich zugänglich sind, sind diese doch aufgrund der überwältigenden Datenmengen unüberschaubar, so dass die Arbeitsuchenden nur schwierig in der Lage sind diese zu überblicken oder die passende Stelle für ihre Fähigkeiten und Anforderungen zu ermitteln und zu bewerten.

Gerade die Analyse von Stellenangeboten sowie die Optimierung des Matching-Prozesses einerseits aber auch die Nutzung der vorhandenen Chancen von Big Data und technologischer Weiterentwicklungen andererseits sind von größter Bedeutung, um einen neuartigen Ansatz im Job-Matching zu verfolgen.

Die vorgelegte Dissertation setzt Lösungen ein, die aus Daten (im Gegensatz zu Regeln) lernen, um auf diese Weise besser auf einen sich ständig ändernden Arbeitsmarkt hinsichtlich der Zusammenführung von Arbeitssuchenden und offenen Stellenangeboten reagieren zu können. Angewendete Methoden um diese Herausforderungen zu bewältigen sind Techniken aus dem Bereich des Maschinellen Lernens, welche eine Analyse und Verarbeitung von datenintensiven Sammlungen realisieren und ein besseres Matching zwischen Arbeitsuchenden und Anforderungen in Stellenangeboten ermöglichen. Hierzu wurde erforscht wie Daten aus Online-Stellenausschreibungen, Beschreibungen von Berufsbildern, Lebensläufen, Selbsteinschätzung des Arbeitssuchenden und Daten aus sozialen Netzwerken analysiert und für ein besseres Matching zwischen Stellenangeboten und Arbeitsuchenden eingesetzt werden können. Eine spezielle Form des maschinellen Lernens ist die Methode des „Deep Unsupervised Feature Learning“. Diese wurde dazu eingesetzt ein neuartiges bidirektionales Matching von Arbeitsuchenden und Stellenangeboten zu entwickeln, indem es ein Modell mit Hilfe von Daten aus Selbsteinschätzung, Resumé-Parsing und sozialen Netzwerken modelliert. Zusätzlich erfolgt ein Parsing der Stellenbeschreibungen und eine Anreicherung mit allgemeinen Anforderungen auf Basis der Berufsbilder. Die Wahl der Daten beruht auf ihrer Eignung verschiedene Aspekte des Arbeitsuchenden und der offenen Stellen zu modellieren. Maschinelles Lernen als Analysemethode wird u.a. aufgrund der Dynamik des Arbeitsmarktes gewählt, da sich Arbeitsanforderungen so häufig ändern, dass die Entwicklung von Regeln praktisch unmöglich ist, während das Lernen aus Daten durch die Verfügbarkeit von großen Online-Datensammlungen und robusten Rechenressourcen möglich ist.

Die Ergebnisse der Forschungsfragen sind i) die Entwicklung eines kontext-sensitiven DTF-Algorithmus für Benutzerprofil-Umfragen; ii) eine neue Technik zur Messung der Qualifikationsrelevanz durch die Modellierung von Arbeitssuchenden mit Hilfe von Daten sozialer Netzwerke; iii) ein verbessertes Relevanz-Ranking von offenen Stellen durch eine Feature-Anreicherung mittels Berufsbezeichnung/Jobtitel und Beschreibungen aus Berufsbildern.

Contents

List of Figures

List of Tables

List of Acronyms

AI	Artificial Intelligence
ANN	Artificial Neural Network
CASCOT	Computer Assisted Structured Coding Tool
CLIR	Cross-Lingual Information Retrieval
CNN	Convolutional Neural Networks
DNN	Deep Neural Networks
DOT	Dictionary of Occupational Titles
DTF	Dynamic Text Field
ESCO	European Skills, Competences, Qualifications and Occupations
HCI	Human-Computer Interaction
HRM	Human Resource Management
ICT	Information and Communication Technology
IDF	Inverse Document Frequency
ILO	International Labour Organization
IoT	Internet of Things
ISCO	International Standard Classification of Occupations
IT	Information Technology
KldB	German Classification of Occupations

KM Knowledge Management

MIT Massachusetts Institute of Technology

ML Machine Learning

NAICS North American Industry Classification System

NIOCCS NIOSH Industry and Occupation Computerized Coding System

NLP Natural Language Processing

O*NET Occupational Information Network

PII Personally Identifiable Information

RDBMS Relational Database Management System

SVD Singular Value Decomposition

TF Term Frequency

UI User Interface

VSM Vector Space Model

Dedicated to my lovely better-half Aynalem Tesfaye and my kids Edna and Mathias who tolerated me when I did not give them the paternal time they deserved.

Chapter 1

Introduction and Background

Bidirectional job matching of vacancy to job seeker involves decision to whether or not a job vacancy is relevant, given the profile of job seeker and vice versa. Greenberg (2010) defined job matching as "the process of matching the right person to the right job based upon the individual's inherent motivational strengths. It requires thoroughly understanding the job and the person under consideration."

Achieving job matching that serves the expectation of job seekers and recruiters has been a challenging task in recruitment industry. This is due to limitations in job seeker and vacancy data that pose difficulties in processing and understanding job seekers, vacancies, or matching process (Furtmueller et al., 2011). These problems are caused by difficulties in obtaining job seeker and vacancy information due to job seeker inability or resistance to provide sufficient data about him/herself (Belloni et al., 2016), difficulty in modeling job seekers and/or vacancies and the complexity of matching process itself.

Availability of large volume of data – which are already in electronic form and ready for analytical use – and predictive analysis techniques helps users to make better decisions, take more consistent actions and reduce costs. This study demonstrated how web mining, natural language processing and deep learning can be combined to achieve bidirectional matching of job seekers to vacancies using features extracted from textual data sourced from the web.

After discussing the theoretical foundations and practical applications in existing literature, this study extends the current body of knowledge in the field of job matching. It does so, by developing a framework that utilizes machine learning techniques in job seeker to vacancy matching using multifaceted data.

Machine Learning (ML) is a study that is concerned with the question of how to construct computer programs that automatically improve with experience. Machine learning, precisely, is described as a "computer program [that learns] from experience E with respect to some class of tasks T and performance measure P, if its performance at tasks in T, as measured by P, improves with experience E. This definition could be used as a design tool to help us think clearly about what data to collect (E), what decisions the software needs to make (T) and how we will evaluate its results (P)." (Mitchell, 1997)

In order to perform matching vacancies to job seekers, this study utilized various data about both job seekers and vacancies. On one hand, it utilizes job seeker resumé, web survey, social networking to analyze and model job seekers. On the other hand, it uses online vacancy advertisements and occupational standards data for vacancy analysis and modeling.

This chapter provides an overview of the thesis by establishing the background and context, highlighting the problems, stating the questions and goals pursued by this research together with their relevance and significance. It also introduces the methods employed, highlights the contributions of this research to the current body of scientific knowledge and ends by laying out the structure of the dissertation.

1.1 Background

Emergence of Information Technology (IT) resulted in generation of huge amount of data every second. Data is collected from diverse sources using different tools and methods such as web forms and documents. It can also be collected from different devices that are fitted with sensors that produce data. This overwhelmingly large data contains hidden knowledge that can be useful for improving organizational strategies in many ways such as designing and developing innovative products, services or processes in organizations (Hislop, 2013, Soto-Acosta et al., 2016). With the availability of large volume of data which are already in electronic form and ready for analytical use, predictive analysis techniques helps users to make better decisions, take more consistent actions and reduce costs. Thus, the need for complex data analytics – which involves the application of advanced analytic techniques to very large, diverse data sets that often include varied data types – will be of paramount importance.

Findings of a single discipline may not hold valid on its own in the view of the real world problem that demands integrated systems to solve it. For this reason, various researchers in different disciplines

are working on identifying parameters, fine-tuning values to reach at optimized results. Job matching is one such problem that demands multidisciplinary effort to achieve reasonable result (Manroop and Richardson, 2016, Persch et al., 2015). Eduworks (Eduworks, 2014) is a project dealing with variables, assumptions, values and findings pursued by multiple independent researchers from various disciplines – labor economics, sociology of occupation, human resource management, lifelong learning and knowledge management – focusing on labour market dynamics to address the education-person-job matching. This matching process takes as input the job requirements by analyzing vacancy data to determine the skill set requirements and specifications of jobs; the abilities and profiles of individuals to determine the skill set acquired by potential employees to be able to uphold the job specified; matching the job vacancy requirements to his/her skill set; and suggesting a certain skill development strategies to fill the identified skill gaps.

In the frame of Eduworks (Eduworks, 2014), this research is primarily intended to apply knowledge based analysis techniques using the integration (Dong and Srivastava, 2013) of data from diverse sources to build a web portal that enables a unified view of different perspectives in person-job matching. The integrated parameters from these diverse and multidisciplinary data gives insight – in relation to various seemingly unrelated parameters – that reflect the holistic view to help smart decision-making (Manroop and Richardson, 2016).

1.2 Problem Statement

It is becoming customary that recruiters look for online profiles of potential employees from professional networking sites. Employers as well as job agents publish job vacancies and produce large volume of vacancy data online which can be exploited to portray the current demand of the job market (Heath and Bizer, 2011). Growth in online recruitment (Russell-Rose and Chamberlain, 2016) has motivated job seekers and job holders to produce huge online jobseeker data through entering the description of their current job, skills they possess and their preferences (Mytna Kurekova et al., 2014).

A significant number of jobs are not filled because of lack of the right applicant (Belloni et al., 2016, Santos, 2016). Although vacancies are publicly available online (Mytna Kurekova et al., 2014), because they are in overwhelming volume, it is not easy for job seekers to find relevant vacancy for their skill. For example, as shown in Figure 2.4, the search for the position of "Internet of Things" returned over

7000 results, of which top-six entries, namely: *Java Software Developer for Internet of Things*, *Software Developer for Internet of Things*, *Senior Research and Development Engineer Automotive Internet of Things*, *Junior Operator - Cloud Computing, Mobile Solutions, Internet of Things*, *Customer Project Manager - Market Segment Industrial Internet of Things*, and *Senior Consultant for Internet of Things/ Industry 4.0* are highly variable (StepStone, 2016). From this result, it is clear that the volume of the total search result, i.e., 7438 entries (cf. Figure 2.4) and the diversity of vacancies that are presented in the top-six list make it difficult for an applicant to get the right job. Due to this volume, not all vacancies are reachable by job-seekers who have relevant skill set, which, in turn, lets unsuitable job seeker to land in the job while the suitable ones are left out.

The problem is partly because, not all vacancies specify all the required skills. Moreover, vacancies are not often prepared with desired skill sets included in the requirements despite the fact that employers consider them in the recruitment and selection process. Job seekers whose jobs are not up to the expectation of their skill set and preferences keep looking for dream jobs. Employees' job descriptions are prepared independently of job vacancy requirements and stored online including in professional networking sites, i.e., they are not tailored to the specific vacancies.

Literatures confirm that challenges in matching of job seekers with relevant skill set to vacancies play significant role in employers' or recruitment professionals' inability to get the right employee for the job requirements, on one hand, and difficulty of job seekers to find jobs that fit their skills, preferences and expectations, on the other hand (Cedefop, 2014, Şahin et al., 2014a). This limitation in matching is either due to limitations in vacancies, job seekers and/or matching vacancies to job seeker, or a combination of two or more of them. Job descriptions in vacancies are often not complete, i.e., employers fail to stipulate some information in either essential or preferred requirements or both during vacancy announcement. This incomplete information makes the vacancy matching process to produce undesired result, e.g. matching job seekers who fulfill the preferred requirements but not the essential ones who will then poorly perform on the job as they lack the essential requirements. The jobseeker side challenge for job vacancy matching is when applicants do not have complete information through under- or overstated qualification, skills and preferences that affects their suitability to the job during matching. These job seeker-to-vacancy matching motivations (i.e., availability of data about job seekers and vacancies) and challenges (i.e., employee-job mismatch, incomplete data in vacancies and job seekers) lead us to the research questions raised in Section 1.3.

1.3 Research Question

Optimized organizational workforce begins with filling job positions with employees who are capable and effective in contributing towards the corporate goals from the start and get engaged for longer period. The first step is attracting talented people in the marketplace who have the requisite skill set. The next is matching their preferences with the advertised vacancy that will not only encourage them to come aboard, but also to make them stay long term. Hence, the main question that this research tries to address is: How can the available multi-disciplinary data about vacancies and job seekers be collected, integrated, represented and manipulated in such a way that the holistic view of the knowledge extracted out of this synergy be used for job matching all automatically?

More specifically, it focuses on the following issues that emanate from splitting this question into sub questions:

- How can we integrate and represent the data from various data sources for use in job seeker and vacancy analysis and modeling?
- How does the demand expressed by vacancies be matched with demand expressed by characteristics given by job seekers?
- How precisely can we recommend vacancies, given job seeker's skill sets?
- How can we implement a better user interface for occupational data collection that employs dynamic text field?
- How can we incorporate social-network-generated data into job seeker modeling for better matching with vacancies?

1.4 Research Goals and Objectives

In order to answer these questions set in 1.3, goals of collection, integration and representation of vacancy and job seeker data are important. This research which is within the framework of the (Eduworks, 2014) project that is a multidisciplinary, and interestingly challenging research which requires amalgamation

of socio-technical expertise in one package, i.e., the study involves social networking, labour economics, human resource management, knowledge management and machine learning. It has the broad objectives of job matching and user interface related improvements. Thus, the main objectives of this research involve improving user experience (Garrett, 2010) in order to collect job seeker skills and preferences (Ferrara et al., 2014) and enrich the job seeker information with social networking data (Mayer, 2011). More specifically, i) exploration of state-of-the-art human computer interaction in web based system development to improve user interface for collecting job seeker skills and preferences; ii) design and implementation of a prototype web based user interface aimed at improving user experience during data collection from a job seeker; iii) collection and analysis of online vacancies and social networking data, and representing and storing them in vacancy and job seeker profile databases, respectively; and iv) modeling a job seeker, vacancy and matching them as depicted in Figure 1.1.

Improvement in the front-end (i.e., user interface) is made by systematically investigating opportunities and challenges in the use of web forms to enhance user experience (Garrett, 2010) for job seeker profile survey through the development of prototype web forms that employ context-aware dynamic text fields (Albert and Tullis, 2013).

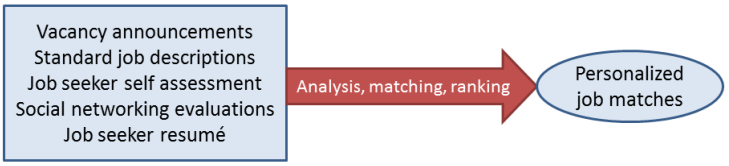


FIGURE 1.1: Generalized view of matching vacancies to job seeker

The design in the back-end (i.e., database) involves two things. First, the development of the underlying database for occupational titles (European Commission, 2015, ILO, 1997) is done. This is used for an exploration of the requirements to the underlying database with more than 1,700 occupational titles (Wageindicator Foundation, 2016). Second, the design of how the data entered by users whose entry is not guided by the dynamic text field will be handled is done. This is used for the development of a procedure of how respondent-side newly added occupational titles, derived from the web-survey, are to be classified in the database.

1.5 Significance and Contributions of the Research

This research stretches discipline boundaries and tackles fundamental problems that need integration of more than one discipline in order to develop person-job matching solution in an interdisciplinary setting. The three core foci of this research are i) modeling job seekers, ii) modeling vacancies and iii) bidirectional matching of job seekers to vacancies through utilization of methods that facilitate data collection, knowledge discovery and presentation. Knowledge discovery is a process of automatically searching large volumes of data for patterns that can be considered knowledge about the data (Maier, 2007). A number of studies are scrambling on the subject of vacancy-to-job matching (Arcaute and Vassilvitskii, 2009, Bhat, 2014, Cedefop, 2015, Greenberg, 2010, Hall and Schulhofer-Wohl, 2015, Mang, 2012, Morgan, 2008, Persch et al., 2015). This is partly because it is highly dynamic, expensive, and multi-disciplinary; and partly because problems in job matching have not yet been precisely addressed.

The research also addresses a problem of major financial significance. As highlighted by Russell-Rose and Chamberlain (2016), the potential user of this bidirectional matching system, recruitment industry, has estimated annual worth of £30 billion in the EU (Tait, 2014) and close to $100 billion in the US (Lupu et al., 2014). The industry also has a turnover of €133 billion every year. The international confederation of private employment services reported that, in the year 2015 alone, the global recruitment industry got 8.6% annual growth with sales revenues of over €450bn (International Confederation of Private Employment Services, 2016). The report also reveals permanent recruitment delivered 11% of revenues over €48bn, while recruitment process outsourcing represents 6% of sales at €27.2bn, and other human resource services brought in 13% of revenue (International Confederation of Private Employment Services, 2016) as shown in Table 1.1.

Yet, studies on regions with high value industries (i.e., Europe, US, China and India) show that clients have a significant level of dissatisfaction, 76% on recruitment companies stating that they did not get value for money (Lupu et al., 2014). In order to cope with how the necessity to study vacancy data has increased over time, "recruiters need to improve their performance to match the expectations of their clients if they are to secure high value placements" (Russell-Rose and Chamberlain, 2016).

The benefits of this system are twofold: i) it helps job seekers by filtering, ranking and presenting job vacancies based on the user profile and ii) it helps employers or recruitment agencies (i.e., those who

Category	**Scope**	**Volume**
Financial Significance	Recruitment worth (EU)	£30 billion
	Recruitment worth (US)	$100 billion
	Annual turnover	€133 billion
	Global Growth	8,6% (€450.4 billion)
Employment and Recruitment Revenue Distribution (2015)	Agency works	70% (€316.6 billion)
	Permanent Recruitment	11% (€48.2 billion)
	Recruitment Outsourcing	6% (€27.2 billion)
	Other HR Services	13% (€55.6 billion)

TABLE 1.1: Recruitment industry financial worth and annual turnover (International Confederation of Private Employment Services, 2016, Lupu et al., 2014, Tait, 2014)

publish job vacancy announcement) in selecting, filtering, ranking and presenting the job seekers that fits to the particular job vacancy.

This research uses historical data for analysis and, hence, the findings reported here are limited to existing data. That is, future trends are not incorporated and are not reflected in the findings. Nevertheless, the implementation of this system enables future data collection for the identified parameters and perform analysis in real-time to update the content with continuous data collection through Web mining and the current situation will be reflected. This enables decision makers to base their conclusion on the most-up-to-date data and solid ground, and hence increases the likelihood of success.

Contributions of this research to the current body of knowledge is improvement in relevance of vacancy recommendation and user experience. More specifically, the contributions of this research are both practical and theoretical. The practical contributions are i) development of an algorithm and a prototype system usable by both job seekers and employers; ii) better and easier user experience vis-a-vis human computer interaction through Dynamic Textfield (DTF)-enabled user profile survey; and iii) a remarkable improvement in job seeker-to-vacancy matching.

Theoretical contributions, on the other hand are i) development of algorithm for context-aware DTF for user profile survey; ii) a new technique to measure skill relevance using social network-enhanced job seeker modeling; iii) improved relevance ranking of job vacancies through feature enriching by job titles and descriptions from standard occupations; iv) discovery of emerging occupational titles through

clustering from job seeker-to-vacancy matching. During clustering outlier vacancies, i.e., vacancies that could not be member of any existing clusters hint emerging occupational title.

1.6 Methods and Procedures

This section briefly describes of the general system overview, components in the conceptual framework, the methods and phase-wise work- and data-flow in the developed system. The implementation aimed at addressing information overload by extracting knowledge (Doan et al., 2009) from data to support users in data entry and search for matching vacancies. The general framework of the system depicted in Figure 1.2 describes the model of job vacancy recommendation enhanced by i) user interaction with dynamic text field in web-based data collection system, ii) social network analysis of user profile and iii) bidirectional matching of job vacancies with occupational standards.

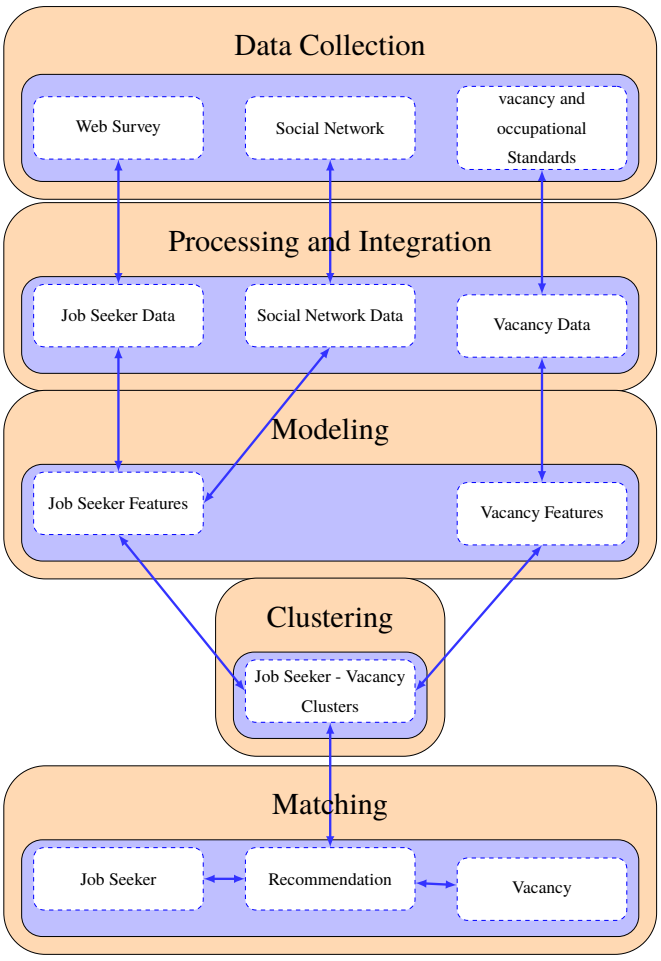


FIGURE 1.2: Overview of the procedure and components

In the system that involves big data analytics, data from various sources (such as the Internet and partners of EDUWORKS (Eduworks, 2014) project) is integrated in such a way that it suits the implementation of the bidirectional person-job matching system. Bidirectional matching is employed in two ways: i) through matching of vacancy advertisements with corresponding job description from occupational standards with the objective of enriching vacancies, which in turn helps improve its relevance when matching the vacancy to job seekers ii) through matching job seekers to vacancies and vice versa. As shown in Figure 1.2, the system is composed of modules for data collection, preprocessing and integration, modeling, clustering and matching (cf. Chapters 4, 5 and 6).

For both job seeker and vacancy, the process begins with collection of data from various sources. That is, vacancy data is collected from online open access sources through web mining, and job descriptions and occupational standards from European Skills, Competences, Qualifications and Occupations

(ESCO) (European Commission, 2015). Similarly, web survey data for job seeker is obtained from WageIndicator (Wageindicator Foundation, 2016) – partner organization; social network data is fetched from StackExchange (Stack Exchange Inc., 2016).

Using DTF enabled web survey for input recommendation when collecting job seeker information, additional data about job seeker could be obtained via the web interface. The advantages from inclusion of this method are i) ability of a user interface to adapt to context and ii) it is ergonomically suitable and thus enhances user experience with the system.

In order to model the job seeker and vacancies, the collected data needs to be preprocessed. Preprocessing involves converting data to plain text, cleaning up the impurities such as white spaces and other noisy characters; removing html tags, punctuation marks; converting the data encoding to universal format to make it ready for training of the system; and integrating the data into a unified format.

After data is cleaned and integrated, job seeker and vacancy modeling step is performed through feature selection (Li et al., 2016) and representation. A feature is a value of an "individual measurable property of a phenomenon being observed" (Menzies et al., 2016). For example, images have pixels, speech has spectra while text documents have words as features (Deng et al., 2014). Feature selection is identifying a subset of the most important features that produces comparable results as the original entire set of features (Li et al., 2016, Song et al., 2013). Feature selection helps to improve prediction performance, reduce computation time, and provide a better understanding of the data in machine learning or pattern recognition applications. Job seeker and vacancy data are clustered based on the similarity of their features so that intra-cluster job seeker to vacancy matching is performed.

Bidirectional matching measures the degree of semantic similarity of job descriptions against vacancies and then provides feedback to improve job descriptions as well as feed-forward suggestions for the improvement to the preparation of accurate vacancies. The added values of this method are: i) improving content of vacancies, and ii) identifying new occupational titles. In addition, matching is done bidirectionally, i.e., for every job seeker, relevant vacancies are searched, filtered, ranked and presented. Likewise, for every vacancy, relevant job seekers are searched, filtered, ranked and presented.

1.7 Structure of the Dissertation

This dissertation is structured into eight chapters.

Chapter 1, which is being closed here, introduces the research, provides the background and overview of the scope. It also states the problem, the questions, goals and objectives this research is aimed at achieving. Moreover, Chapter 1 highlights the methods and procedures employed as well as significance and contributions of the research.

Chapter 2 explores the state-of-the-art of the extant literature as well as the current ones with the overarching objectives to i) investigate the different techniques in job seeker modeling and their challenges, ii) investigate different techniques of vacancy analysis and modeling, iii) investigate factors that affect the quality of the job vacancy recommendation, and iv) study how job vacancy recommendation improves the job seeker as well as employer experiences in person-job matching chores.

Chapter 3 begins the theoretical and conceptual dimensions of the research, and looks at relevant concepts vis-á-vis how job vacancy recommendation systems are implemented as well as their theoretical backgrounds.

Chapter 4 covers in-depth exploration of variables related to job seeker modeling that affect job recommendation, algorithms as well as technologies employed for job seeker modeling. Likewise, selection, filtering and ranking of job seekers in order to match it with a job vacancy is also explained in the chapter.

Chapter 5 describes variables related to job vacancy modeling that affect job recommendation, algorithms as well as technologies employed for job vacancy modeling.

Chapter 6 explains the task of matching job vacancies to job seekers. The chapter explains how job vacancies are selected, filtered, ranked and presented to the job seeker. Moreover, algorithms as well as technologies employed for matching and recommendation are discussed.

Chapter 7 demonstrates the experimental setup of the research vis-á-vis the prototype system with the data used as well as the procedures for the experiment. The chapter also discusses the findings of the research, evaluates and explains the results.

Finally in Chapter 8, conclusion of the study are highlighted and future research directions are outlined.

Chapter 2

Review of Related Works

Job matching has been a challenging task in the recruitment industry due to difficulties in processing and understanding job seekers, vacancies, or matching process (Furtmueller et al., 2011). The problems are associated with i) lack of information due to job seeker inability or resistance to provide sufficient data about him/herself (Belloni et al., 2016), ii) difficulty in modeling job seekers and/or vacancies and iii) the complexity of matching process itself. There have been proposed solutions such as applying ontology mapping (Senthil Kumaran and Sankar, 2013). These solutions have fallen short in dealing with the dynamics of job market, i.e, the jobs change so frequently that developing rules is practically infeasible. As a result, solutions that learn from data (as opposed to rules) weigh better at handling job seekers and vacancy data in the ever changing market. One of the methods to address these challenges is applying Machine Learning – data-intensive techniques to model job seekers and vacancies – and get a better matching. This research explores the study on matching job vacancies with job seekers using data from online vacancies, occupational standards, job seeker's self-assessment, and social network analysis.

This chapter explores and reviews literatures on problems and methods of data-intensive occupational knowledge based systems, vacancy recommendation, and human computer interaction with the objective of elucidating the gap in the state-of-the-art for job seeker to vacancy matching, which is the thesis of this research. More specifically, the chapter highlights reviews of literatures on occupational information systems, knowledge based methods utilized in analyzing, modeling and matching of job seekers with vacancies, and provides an overview of existing online job recommendation systems. It reviews literatures on various aspect of job seeker to vacancy matching and discusses challenges in job matching and

foundations in occupational information systems. It summarizes approaches attempted to address these challenges in relation to job seeker and vacancy analysis as well as matching processes. To shade light on this context vis-á-vis existing applications, the chapter also presents a review of a couple of online job matching systems and draws conclusions on their features as well as limitations.

Job matching has already been studied for a while in various fronts in different fields from a number of perspectives. This section explores studies that are similar to this one in their focus, scope and/or methods used with the goal of pinpointing the commonalities and uniqueness of this research. This review aims at setting the foundation for modeling job seekers and vacancies, and enriching them from social networking and occupational standards, respectively. It presents review of literatures on occupational information systems, dynamic interface for job seeker data collection, social network analysis for job seeker modeling, online vacancy mining and modeling, and job seeker and vacancy matching.

2.1 Occupational Information Systems

Occupational information system is a system that contains a collection of occupational definitions that help job seekers, businesses and recruitment agencies to understand the specifications and requirements of an occupation. Occupational information systems describe occupation in terms of the skills and knowledge requirements and how the work is performed in a typical work setting.

The occupational information systems considered for review in this study are: Dictionary of Occupational Titles (DOT) (Coutsoukis, 2011), Occupational Information Network (O*NET) (Peterson et al., 1999), North American Industry Classification System (NAICS) (US Census Bureau, 1997), NIOSH Industry and Occupation Computerized Coding System (NIOCCS) (CDC, 2016), German Classification of Occupations (KldB) (Paulus and Matthes, 2013), International Standard Classification of Occupations (ISCO) (ILO, 1997), ESCO (European Commission, 2015), and Computer Assisted Structured Coding Tool (CASCOT) (Jones and Elias, 2005). These eight occupational information systems were analyzed using five features as criteria for evaluation of user interaction with web forms (Hillier, 2002); (Trewin et al., 2010); (Werner and Fulton, 2012) as summarized in Table 2.1. Analysis of features of these occupational information systems (ESCO (European Commission, 2015), ISCO (ILO, 1997), KldB2010 (Paulus and Matthes, 2013), DOT (Coutsoukis, 2011), NAICS (US Census Bureau, 1997), CASCOT (Jones and Elias, 2005), O*NET (Peterson et al., 1999), and NIOCCS (CDC, 2016)) vis-a-vis their User

Interface (UI) has been done using five indicators pointed out in (Hillier, 2002); (Trewin et al., 2010); (Werner and Fulton, 2012) to evaluate user interaction with web forms, namely, whether they i) are web-based, ii) used any UI elements and if they used open-ended or close-ended or autocompletion in their UI, iii) support multiple language, iv) are adaptive (have learning facility), and v) have in-built analysis system as summarized in Table 2.1.

The occupational information systems shown in Table 2.1 are presented in five criteria of evaluating the systems, namely: *Web-based*, *UI Elements*, *Language Support*, *Adaptive* and *In-built Analysis*. The *Yes* in *Web-based*, *Adaptive* and *In-built Analysis* columns denotes the presence of the feature while the *No* denotes the absence. For *UI Element*, the *Close* is used to show that the system uses close-ended UI elements, the *Open* represents that the system uses open-ended UI elements, *Open+Close* denotes the use of both open- and close-ended elements, whereas the *NA* (i.e., not applicable) shows that the system uses neither close-ended nor open-ended UI elements. This describes the occupational information systems which are predominantly manual. The *Language Support*, as reflected in its name, represents the languages supported. The entries *EN*, *DE*, *FR* and *Multi* stand for English, German, French and multiple languages, respectively.

Systems	**Web-based**	**UI Elements**	**Language Support**	**Adaptive**	**In-built Analysis**
ESCO	Yes	Close	Multi	No	No
ISCO	No	NA	EN	No	No
KldB2010	No	NA	EN/DE	No	No
DOT	No	NA	EN	No	No
NAICS	No	NA	EN / FR	No	No
CASCOT	No	Open+Close	Multi	Yes	No
O*NET	Yes	Close	EN	No	No
NIOCCS	Yes	Open+Close	EN	No	No

TABLE 2.1: Review of existing occupational information systems

As shown in Table 2.1, only three out of the eight systems that have been assessed – ESCO, O*NET and NIOCCS – are web-based; two of them – ESCO, O*NET – use entirely close-ended UI elements and the others – CASCOT and NIOCCS – have a mix of open- and close-ended ones. While all the eight

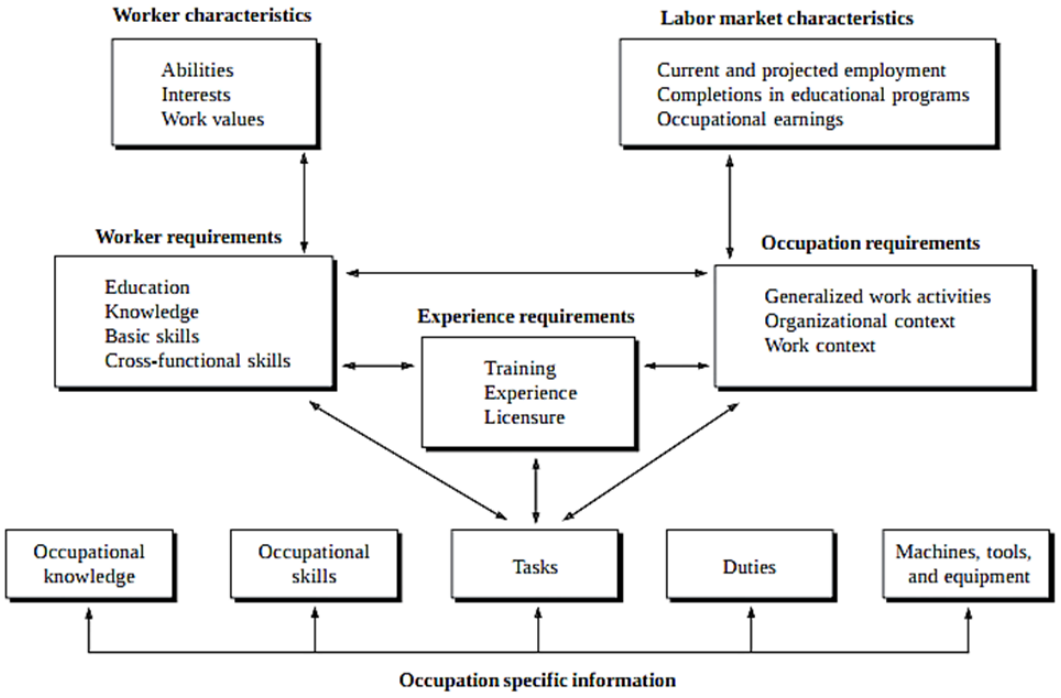


FIGURE 2.1: The O*Net Content Model (Adapted from Mariani (1999))

systems are in English, two of them – KldB2010 and NAICS – are bi-lingual, i.e., German and French respectively, in addition to English, and two of them – ESCO and CASCOT – are multilingual.

Analysis of user interaction in the existing occupational classification systems – DOT, O*NET (Peterson et al., 1999), a number of potential improvements have been identified. The first system is a manual system. Its creation and update is entirely manual, and therefore, is tedious and error prone when performing retrieval. In addition, it is not automatically updated in case a new occupational title arises.

O*NET (Peterson et al., 1999), on the other hand, is automatic system which alleviates the problem of slow retrieval that exists in DOT. Content specification in O*NET, as shown in Figure 2.1, combines workers' details, labor market requirements, occupational requirements, job specific information and provides a web-based interface. However, it has a number of limitations. First, it is based on static data entry. As such, it is not based on actual data or existing knowledge. Second, its user interface utilizes either open format data entry elements such as a textbox or closed format data entry elements such as list box. Thus, it does not have dynamic interaction between users. This, in turn, results in storing occupational titles that actually exists but with a different diction.

CASCOT is multi-lingual as it supports eight languages (Dutch, English, Finnish, French, German, Italian, Slovak and Spanish), despite it not being web-based. While CASCOT is apparently multi-lingual, it only has represented ISCO titles in multiple languages rather than mapping the occupational titles in one language to another countries' occupational titles, i.e., it is multi-lingual but not multi-country.

The occupational classes of ISCO have been taken one level deeper by ESCO, which is a web-based system that utilizes close-ended user interface elements (i.e., hierarchical tree and list) to browse into the system. While it supports multiple languages (the official languages of the European Union except Irish), it lacks adaptability and in-built analysis.

None of the aforementioned systems incorporate a learning component. As a result, the systems will require intervention on the part of the developer for an update to reflect the newly entered data from users. Furthermore, none of them have an in-built analysis feature.

2.2 Dynamic Interfaces for Job Seeker Data Collection

Matching jobs seekers to vacancies would be a lot better if a system could gather information from job seekers in a painless manner either from job seekers themselves or from other sources like social media or their combination. This research explores related scientific studies and implementations on web-based data collection in general and the types of user interface elements employed in the design of web survey instruments to ease data collection from job seekers.

Literatures have highlighted the benefits (Evans and Mathur, 2005) and challenges (Couper and Miller, 2008, Tuten et al., 2000) of data collection in general and online data collection (also known as web surveying) in particular together with strategies of addressing them. These studies found out that online data collection has a number of sociological/psychological, technological, and economical challenges. In addition to studies on challenges and recommended strategies, various design and implementation efforts have been made toward facilitating web-based data collection by addressing these challenges. The focus of this research is the mix of sociological/psychological (the human factor) and technological (the design factor) challenges and a particular strategy (i.e., assisting users in the process of data collection through autocompletion) to address them.

Couper and Miller (2008) found out that web surveys have contributed significantly to survey research, in spite of their relatively short history. Tuten et al. (2000) argued that web surveys are inexpensive because they reduce the amount of printing, data entry and logistical costs associated with more traditional survey measures. They are also convenient to provide data ready for automated processing without further delay after data collection (Tuten et al., 2000). Because the user (as opposed to a data entry clerk) inputs the data directly into the data file that will be fed to the processing functions without the need of any third party involvement (Tuten et al., 2000).

Evans and Mathur (2005) discussed a number of strengths and potential weaknesses of online survey research. The most notable of the limitations that are manifest to Human-Computer Interaction (HCI) are lack of online experience/expertise on the part of respondents and unclear instructions to answering the questions (Evans and Mathur, 2005). They included in their suggestion that researchers need to assist respondent(s) through various techniques and "make it effortless to enter answers" Evans and Mathur (2005).

The methods to choose which user interface elements to implement for data collection instruments has been studied extensively in the last couple of decades. The most notable ones are: Reja et al. (2003), Schuman and Presser (1979), Dillman et al. (2011), Jansen et al. (2007), Schleyer and Forrest (2000), and Lauer et al. (2013). Their findings sum up the recommendation that usage of open-ended format is more suitable than close-ended format toward tackling two difficult tasks i) when the purpose of data collection is to discover spontaneous responses of individuals or ii) when it is impossible to code answers sufficiently into specific options – both at the expense of the bias introduced from limited options for responses.

Research on using dynamic text field (also known as autocompletion text box or autocomplete) employs techniques used in content-based recommender systems. Occupational information obtained from the use of document analysis and modeling technique that will be used to guide the user through data entry process can be very different depending on the number and types of attributes used to describe them. In addition, many efforts have been made to improve data quality by enhancing the user interface for data entry using autocompletion strategy. Ward et al. (2012) studied the use of autocomplete as a research tool. Khoussainova et al. (2010) studied, implemented and evaluated autocompletion system called SnipSuggest to assist the writing of SQL statements in databases. Bruch et al. (2009) developed an example-based system for source code autocompletion in software engineering. Amin et al. (2009) studied the effect of organizing suggestions and compared different methods of organizing autocomplete

to ease selection process for user to search thesauri systems. Chen et al. (2010) investigated the potential of adaptive feedback (i.e., a probabilistic model that adjusts the user interface based on previous data entry) for improving data quality in clinical research. The most common use of autocomplete is in search engines to fill queries based on previous entries. The notable example is Google® that uses auto-complete to predict user queries based on previous entries Google (2015).

A number of studies have been conducted to develop algorithms to achieve autocompletion. For example, Levenshtein Atomata (Levenshtein, 1966) is the most commonly used one for single-word autocompletion Schulz and Mihov (2002). Troiano et al. (2009) employed Bayesian Networks to predict user entries for autocompletion. Matani (2011) implemented autocomplete using sorted arrays of phrase prefixes and achieved a relatively memory efficient result. Thus, it is not yet conclusive as to which algorithm is the best in satisfying the requirement of data entry through recommendation.

For the works on autocompletion that are performed on natural language (Chen et al., 2010, Google, 2015), the source of knowledge for generating suggestion is the previously entered user queries than the underlying dataset. This is subject to limitations where it fails to suggest if the partial input is too new, or if the topic is not frequent enough to be a candidate in the suggestion list. On the other hand, the works on autocomplete that are performed on formal languages Bruch et al. (2009) and Khoussainova et al. (2010) are good for autocompletion of codes and are thus not suitable for natural language data entry in surveys, e.g., to collect job seeker information.

It is important to keep the balance between limiting the number of choices and giving freedom to the end user. By doing so, the system provides the user options to select from while at the same time he/she is still free to enter what he/she wishes even if recommendations are provided. This is what makes it different from mere autocomplete systems. In DTF, recommendations are produced by analyzing and using the underlying huge amount of data in contrast to systems that use previous queries Chen et al. (2010), Google (2015) to make suggestions on natural languages and as opposed to formal languages Bruch et al. (2009), Khoussainova et al. (2010).

After analyzing these existing systems, a few gaps have been identified, some of which can be addressed by implementing DTF – that interface elements either: i) are human intensive and thus tedious and error prone, ii) do not have dynamic interaction with users, iii) are not automatically updated in case a new occupational title arises, iv) need an entirely manual update of UI for new occupational titles, v) are prone

to storing occupational titles that actually exist but with a different diction, thereby creating unnecessary redundancy, vi) do not support multiple languages, or vii) have no integrated analysis.

To sum up, the first four issues in the list above can be addressed by implementing DTF that improves user experience through i) assisting users by suggesting ranked probable entries for the user to choose. By doing so, it reduces the pressure and potential errors in data entry. ii) It enables dynamic interaction and engaging users in filling the forms. iii) In case, the user does not want to select one of the suggested entries, DTF also allows the new entry to be recorded. iv) As the items in the suggested lists are updated automatically in DTF, there is no need to manually update the interface to include new occupational titles. DTF is implemented using methods and techniques discussed in Chapter 4 in order to address these limitations.

2.3 Social Network Analysis for Job Seeker Modeling

In labour market construct, both job seekers and employers participate in job search - the former looking for vacancies (i.e., job offers) and the latter looking for a candidate employee for a job position. Each party provide what they can offer and what they require, i.e., employer provides details of job description in vacancies together with benefits associated with the position. On the other hand, job seekers provide the outline of their knowledge, skills and competences. That is where the matching problem – determining which job seeker is suitable for which vacancy and vice versa – comes into picture.

A number of studies have stressed the role of social network in matching workers to jobs. Informal connections through social networks have been contributing in provision and acquisition of information in job search process. This has been confirmed by studies that conducted extensive surveys across countries and for extended span of time that have indicated a significant proportion of jobs are filled through matches created by information obtained from network connections (i.e., relatives, friends and/or friends of friends) (Ioannides and Datcher Loury, 2004, Topa, 2011).

Through surveys and models, these studies stress the benefits of social network referrals in matching process with respect to resolving uncertainties in unobserved qualities resulting in higher salaries and lower turnover. The information obtained from social contacts is useful to resolve uncertainties in unobserved qualities for both job seekers and employers. Moreover, Brown et al. (2016) confirmed that social network referrals provide useful information that improve job matching and thereby reduce turnover.

Cahuc and Fontaine (2002) developed a simple model job seekers with employers through social networks. The study used social network to decentralize and increase efficiency in job search process and reported that there is a significant contrast in the search result with regard to intensity and efficiency. However, the study utilized the connection (i.e., know-who) aspect of social network, as opposed to detailed knowledge of the job seekers expertise, skills and preferences.

Cappellari and Tatsiramos (2015) developed an approach useful for estimating social network effects in the labor market using a measure of network quality. Measure of network quality is characterized by employment status of close connections, i.e., job finding rate is positively affected by the number of close friends who are employed.

A number of studies have been carried out on job recommendation systems (Panniello et al., 2014) in general and on application of social network analysis on recruitment in particular. The studies on usage of social networks to facilitate recruitment in the knowledge society, with a case study in Bulgaria, confirmed the role social networks play in supporting recruiters (Toteva and Gourova, 2011).

Athavaley (2007) summarized how the online contacts can be used by recruiting managers to get more information about potential candidates for a job using their online profiles in professional networking sites such as LinkedIn® and Jobster®.

Research by (Pérez-Rosés, Hebert and Sebé, Francesc and Ribó, Josep Maria, 2016) reported a result on using directed graph for endorsement deductions and ranking. They developed an algorithm that adds new weighted arcs to the digraph of endorsements based on the relation of endorsements.

Otte and Rousseau (2002) explored theoretical foundations of social network analysis and practical application of it with particular emphasis to the information sciences. Morgan (2008) conducted study on job preference assessment. This connects how social network analysis and its application can support assessment of job seeker preferences.

Garg and Telang (2011) studies the role of individual's social networks on one's job search behavior. Garg and Telang (2011) also reported on the extent to which the strength of connections affect job outcomes from online social networks and compares the results with that of traditional job searching methods.

In Latkin et al. (2013), social network is used to recruit subjects for testing and counseling through utilization of social diffusion, network stability, choice and training of network members. Only the

relation (as opposed to attributes) aspect of social network is utilized in Latkin et al. (2013). Moreover, Latkin et al. (2013) did not use social networking data to rank the subjects, rather only to reach them.

Melanthiou et al. (2015) investigated the benefits (and challenges thereof) of e-recruitment and the role of social network for this purpose. Melanthiou et al. (2015) identifies companies that utilize social media in the process of recruitment to select and rank job applicants. In addition, Melanthiou et al. (2015) explored the legal issues that are related to social network in recruitment and selection. While discussing the benefits of e-recruitment and the role of social networking in recruitment and selection of job applicant, Melanthiou et al. (2015) did not discuss the specific benefits of social network data in the integrated e-recruitment systems. That is, the attribute and relation data of job seekers are not integrated into the e-recruitment system, rather recruiters used social networking sites to get more information about the applicants.

According to Qian et al. (2013), we can obtain 'attribute' and 'relational' data from social networks. Attributes refer to the data about attitudes, opinions, and other qualities that characterize the participants in the relationship. Relation, on the other hand, refers to the contacts, connections and ties to show structural topology and attachments of individuals in the network.

Although a number of literatures have discussed clear evidences about the importance of social networking data to get more insight about a job seeker, employers have not made extensive use of it, apart from mere exploration of job seeker profile on social networking sites. This is because little was done on integration of social networking data in online job matching and recruitment systems.

The studies reviewed so far have not touched the useful dimension of social network in person-job matching. First, none of them have taken social network data that was generated passively into account, i.e., they use social networking data in active referral for a targeted job that can be highly susceptible to bias.

Second, none of these studies have measured (or utilized a measure) to create an objective metric from social network referrals that can be combined with other evaluation parameters that are used in matching job seeker to a vacancy.

Third, none of these studies have worked on the appropriateness of the job seeker to vacancy matching with respect to skills of the former to requirements of the latter.

In this research both types of social networking data described in Qian et al. (2013), i.e., attributes and relations, are used to model characteristics of a job seeker and the strength of connection he/she has in the network with other peers, respectively.

2.4 Online Vacancy Mining and Modeling

The importance of analyzing online vacancy data for use in employment decision making in recruitment industry has been investigated in a number of researches (European Commission and ECORYS, 2012, Wowczko, 2015).

Mang (2012) conducted a comparative study of job seekers using online job vacancy ads vis-á-vis traditional newspaper-based job vacancy advertisements. The findings in Mang (2012) show that job seekers who use online job vacancy are better matched than the ones that use traditional advertisements – emphasizing the growing importance of online vacancy. The advantages and disadvantages of online vacancy data is explored by Kureková et al. (2015). While exploring the methodological challenges such as research design and representativeness that online vacancy data pose on research, Kureková et al. (2015) appreciated the opportunities associated with exploiting this large data.

With the objective of measuring the effect of mismatch in unemployment in the US, Şahin et al. (2014a) developed a framework that shows how unemployment increases through mismatch between job seekers and vacancies. A study by Capiluppi and Baravalle (2010) shows how employers demand of IT skills in the UK job market affect the content of vacancy websites. In addition to study of whether or not this demand changes through analysis of online job board, Capiluppi and Baravalle (2010) provided recommendation that supports development of curricula to respond to this demand.

Hall and Schulhofer-Wohl (2015) studied a measure of job finding and matching efficiency, i.e., measuring productivity of job matching process for heterogeneous job seekers. Since the framework of matching efficiency developed by Hall and Schulhofer-Wohl (2015) focused only on job seekers, it did not consider the structure of vacancies in the framework.

The Natural Language Processing (NLP) aspect of analyzing content of vacancy data does not seem to be given due attention (Keep and James, 2010). From the literatures reviewed in this research, apart from Wowczko (2015) which utilized machine learning techniques to extract knowledge from textual vacancy

data, all of the studies focused only on supply and demand matching. Using job advertisements as only source of knowledge and with small sample size, Kennan et al. (2006) reported that there is lack of clarity about the skills, knowledge and competencies required.

Very little has been done on automation of online recruitment and a few systems that assist recruiters are available. For example, Scrapit (2017) is a web-based service that collects description of websites from users, scraps the data from the websites that contain the description and extracts the data, and saves the content into spreadsheet so that the users explore or exploit it. Although it saves a lot of time that would be spent copy/pasting data from the web to the spreadsheets, it left the whole process of filtering, matching and ranking to the users.

2.5 Job Seeker and Vacancy Matching

Defined in Merriam-Webster (2016), matching is "to put in a set possessing equal or harmonizing attributes; to cause to correspond; to fit together or make suitable for fitting together." Job seekers to vacancy matching is the process of determining how efficiently workers find new jobs. It helps job seekers to find jobs faster, vacancies filled quickly and for a longer term. It reduces unemployment caused due to job seekers inability to find a suitable vacancy. Matching job seeker to vacancy can also expedite the work of recruitment agencies and enables easy mobility of labour (Nickell et al., 2003).

Bhat (2014) highlighted the relationship between candidate performance with matching jobseeker to vacancy. In support of Bhat (2014), Heathfield (2016) described the relationship of candidate performance in a team with the matching outcome by stating that "without the right job fit, an employee will never experience as much happiness and success as he deserves at work. He'll never achieve his true potential. [. . .] Job fit is a concept that explains whether the intersection between an employee's strengths, needs and experience, and the requirements of a particular job and work environment - match - or not." In addition, Target Training International (2013) formulated ideal candidate form to objectively define the criteria to assess candidate for a job.

Objective definition of features that assess candidates and analyses vacancies is important in order to facilitate automation of data-intensive computations and matching algorithms that utilize NLP and machine learning. Taking into account the challenges of semantic matching such as computational

time, lack of available labeled data and difficulty in feature selection (Christen, 2012), a combination of algorithms needs to be employed for clustering of the textual data, measuring the similarity, matching and searching. Clustering refers to set of methods and algorithms for analysis of objects (e.g. graph, data, document, text or term) to identify related items, and organize them into groups whose members are similar (Biemann, 2012, Fasulo, 1999, Fortunato, 2010, Schaeffer, 2007). Proper selection of a clustering method (or algorithm) is highly related to the application context i.e. clustering of graphs, data, document, text or term. In data clustering (Gan et al., 2007), communities are set of points which are close to each other, with respect to a measure of distance or similarity (Fortunato, 2010). The latter is potentially adaptable distance-based similarity approach (Charu and Zhai, 2012). In the concept of text (or document and term) clustering, there are approaches using hierarchical clustering or text mining methods (Charu and Zhai, 2012). These methods are focused on organizing text data based on similarity or association measure. The approaches are applied in a similar way for document-, text- and term-clustering (Klahold et al., 2014). So the term (word) clustering is used here to refer to such methods. In this context, hierarchical term clustering algorithms (techniques) such as single-link, complete-link, average-link, cliques, and stars (Li, 1990, Rajasekaran, 2005) are applied.

The single-link or single-linkage clustering method detects and merges unlinked pair of points in two clusters with the largest similarity (Manning et al., 2008), while complete-link clustering or complete-linkage clustering determines the similarity of the most dissimilar members of the clusters (Manning et al., 2008). In average-link or average-linkage, the average value of all the pairwise links between points (for which each is in one of the two clusters) is a measure for computing the similarity (William and Baeza-Yates, 1992). The clique clustering groups the data into cliques i.e. identifying subspaces of a high dimensional data space that allow better clustering than original space (Kochenberger et al., 2005).

In addition to the methods discussed earlier, there are a number of related works using ontology-based framework for text clustering (Hotho and Staab, 2002, Ma et al., 2012, Tar and S., 2011, Yang et al., 2008). Based on related definitions in (Guarino et al., 2009), ontology is a "formal explicit specification of a shared conceptualization." Dissecting this definition, we get four key terms that make up the definition: conceptualization, explicit, formal and shared. A *conceptualization* is an abstraction model of a certain phenomenon in the world using the relevant concepts of the phenomenon. *Explicit* refers to the explicit definition of the type of concepts and constraints that dictate the concepts. For example, in job matching, the concepts are job seeker and vacancy. They are related in such a way that job seeker is matched to

vacancy. The constraint can be that vacancy cannot be matched to itself. *Shared* refers to the fact that the ontology represents knowledge that is agreed up on, i.e., it is not individually oriented, but must be accepted by a group. *Formal* representation refers to the requirement that ontology should be in a machine readable format, in which case natural language is excluded.

In string matching, a number of studies have been conducted in the area of both fuzzy and exact matching of patterns (Hussain et al., 2013) in which they applied exact matching algorithm using two pointers (simultaneously) based on window sliding method where they tried to compare bidirectional algorithm's results with Quick Search, BM Horspool, Boyer-Moore and Turbo BM algorithms that are deemed to be efficient for character comparisons and attempts to complete processing of selected text. They used bidirectional matching algorithm that compares a given pattern from both sides, starting from right then from left, one character at a time within the text window and produced an algorithm that scans text string from both sides simultaneously against the given pattern. Its analysis shows that it takes O(mn/2) time where m is the length of the given pattern and n is the length of the target text.

In this study, the approach involves document and term clustering which employs document similarity measures for text clustering, and bidirectional matching that is focused on document matching as opposed to term matching using natural language data. Unlike its usage in (Hussain et al., 2013) bidirectional matching, in this study, refers to matching features (i.e., terms or other values of attributes) in job seekers with job vacancies and vice versa to produce a unified search space for documents in the same cluster.

A number of researches on whether online vacancy data contributes to the study of labor and (un)employment have been conducted. These studies build on the conclusion made decades ago by Dunlop (1966) on measurement of vacancy data. The study related the concerns of vacancy data to theoretical, policy and operational interests that hold to date vis-á-vis understanding the usefulness and thereby utilizing this data (Carnevale et al., 2014). Dunlop (1966) concluded that the job vacancy concepts and data had small role in operations of business enterprises, governments and other labor employers "until these data are perfected and enter into internal organizational processes" (Dunlop, 1966). Despite its limited scope, online job vacancies have unique and rich content that can be exploited to provide useful information to policies in various fields including e-recruitment (Kureková et al., 2013). Moreover, Kureková et al. (2014) ascertained the importance of data collected via web-based individual voluntary survey in representing the sample population's characteristics.

Wagner (2011) studied how jobs are evolving over a period of time because of change in the way organizations perform, by societal dynamics and other factors in the work environments. Moreover, Wagner (2011) also summarized how new skills help job seekers fit into the changing nature of skills demanded in job market as: "retrofitting" - incorporating new skills to existing jobs, "blending" - combining skills sets from different existing jobs or industries to create new specialties, and "problem solving" - the changes in the nature of jobs creates the supply of new problems for people to solve. With vacancies data, that contains the current need of employers, one can get new skills and/or new jobs which have not yet been incorporated in standards. Vacancy data analysis not only helps fill the gap in the data of job descriptions available in occupational standards and job seekers' profiles but also helps identify emerging jobs. Studying the trend of job vacancies and the nature of their changes help device methods of helping job seekers to adapt in the changes.

While mostly focusing on the trend analysis and theoretical researches of job vacancies, previous studies fail short of the usage of actual content in vacancy data to figure out the content-level analysis and to investigate the problem of – and thereby formulate a comprehensive solution to – matching job seeker to vacancy. Moreover, these studies have not taken comprehensive view of the job seeker into account in the matching process.

This bidirectional matching approach combines web-based individual voluntary survey (Kureková et al., 2014) with vacancy data in order to determine the best matching vacancy for a job seeker possessing a particular skill set.

According to the survey conducted by ManpowerGroup (2016), the global percentage of employers who are having difficulties filling job vacancies has risen consistently. For example, the proportion increased from 36% in 2014 to 38% in 2015 and 40% in 2016, making the highest percentage since the global economic recession in 2008 as shown in Figure 2.2.

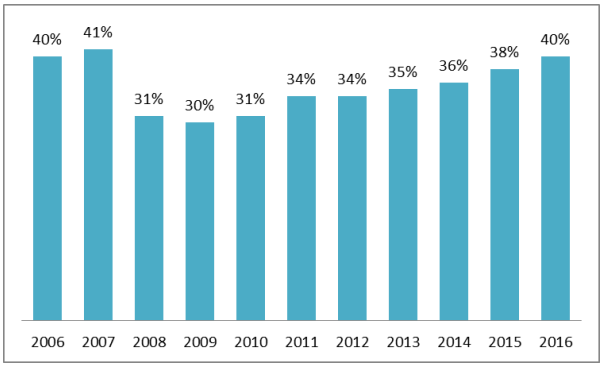


FIGURE 2.2: Talent shortage (Adapted from: ManpowerGroup (2016))

ManpowerGroup (2016) also reported that the most common reason employers mention for the talent shortage is lack of applicant to vacant positions (24%) followed by lack of technical skills (19%) while other factors involving job seeker preferences add up to 33%. The report also found out that, in 2015, IT skill – which is central to Internet of Things (IoT) – is the second hardest to fill. Table 2.2 summarizes the literatures reviewed on job matching.

TABLE 2.2: Consolidated review of literatures on job matching

Reference	**Summary**	**Limitation**
Athavaley (2007)	summarized how the online contacts can be used by recruiting managers to get more information about potential candidates for a job.	not integrated in online recruitment systems
Pérez-Rosés, Hebert and Sebé, Francesc and Ribó, Josep Maria (2016)	researched on using directed graph for endorsement deductions and ranking. They developed an algorithm that adds new weighted arcs to the digraph of endorsements based on the relation of endorsements.	the study assigns weights to relationships of entities in the social network not weighting skills

TABLE 2.2: Consolidated review of literatures on job matching

Reference	**Summary**	**Limitation**
Otte and Rousseau (2002)	explored theoretical foundations of social network analysis and practical application of it with particular emphasis to the information sciences.	has extensive coverage of social network analysis and its application but it stops short of describing the role of social networks in e-recruitment
Morgan (2008)	conducted study on job preference assessment.	performs preference assessment while ignoring other aspects of job seeker
Garg and Telang (2011)	studies the role of individual's social network one's job search behavior, the extent to which the strength of connections affect job outcomes from online social networks and compares the results with that of traditional job searching methods.	highlighted job outcome due to individual's social network and compared the result with traditional job searching without discussing how integration of social network affects job matching

The growth of online recruitment and the significant role it plays in global arena is explored by number of authors. The discussed related works are samples of researches that highlight the use of information technology in analysis of supply and demand in labour market. These researches analyzed digital vacancy data with the prime objective of identifying skill demand in the job market.

2.6 Online Job Matching Systems

There are a number of online job recommendation systems available today. This section provides a review of these systems vis-á-vis their brief description, their data collection, analysis and recommendation

approaches, and the data they utilize. On the grounds of closeness to this study and access, the systems considered in the review are: Beansprock (Beansprock, 2017), IrisJobs (Irishjobs, 2017), StepStone (StepStone, 2016), TextKernel (Text Kernel, 2016), and Venn (Vennjobs, 2016).

Beansprock (Beansprock, 2017) – developed by two Massachusetts Institute of Technology (MIT) graduates in 2013 (Alba, 2015), Beansprock applies NLP and machine learning techniques to match job seeker to a suitable job vacancy.

As shown in Figure 2.3, Beanspock collects job seeker preferences such as company culture and location in such a way that the job seeker selects entries from predefined set. Then the system lets job seeker to profile him/herself if the job seeker is an active seeker or a casual seeker.

In Beanspock, job seekers are required to provide skills out of which they are asked to select the three most important skills. That is, skill weighting, is left to the user and the weights are flat. In other words, the job seeker is forced to decide flat weight to all the skills he/she provides and then re-assigns a flat higher weight to the top three most important once.

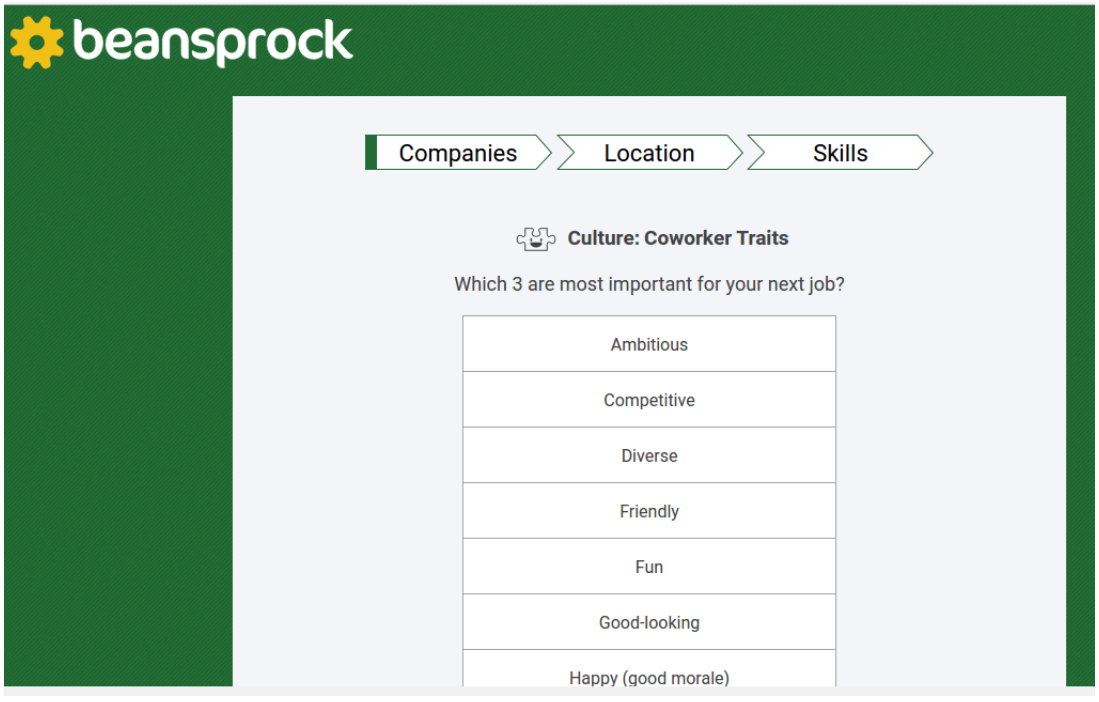


FIGURE 2.3: Job seeker data collection in Beansprock (Beansprock, 2017)

In contrast to Beansprock, this research considers skill weighting as a central criteria to rank skills during matching and provides differentiation between skills through continuous weighting scheme.

IrisJobs (Irishjobs, 2017) provides search features grouping jobs into job advertisements based on recruitment type, roles and skills, location, employment type and salary to facilitate searching. It is primarily a job searching (as opposed to matching) system.

StepStone – Systems such as StepStone (2016), recommend jobs to job seekers (i.e., their subscribers) using a few parameters. For example, as evidenced from the sample recommendation from StepStone shown in Figure 2.4, the system recommends jobs using location and job title and/or parameters alone. This leads to recommending irrelevant jobs to subscribers as clearly shown in Figure 2.4. For this reason, job seekers who are interested in jobs in a particular geographic location (i.e., Siegen in the case of the sample recommendation shown in Figure 2.4) get same set of job advertisements irrespective of differences in their qualifications, experience or preferences, of course, except location. Moreover, the job advertisements recommended to the job seeker are not ranked in whatsoever criteria. In this example, assuming the job seeker is interested in Sales job, the precision of this system is only 16.7%, i.e., only one out of six is relevant, yet it is in the fifth place (cf. Figure 2.4) due to the fact that it fetched more jobs than necessary for this job seeker.

FIGURE 2.4: Job Recommendation by StepStone (StepStone, 2016)

TextKernel – The system developed by (TextKernel, 2015) uses the resumé text of the candidate's profile and automatically creates a search which is performed on multiple and multi-lingual sources of jobs. The search system collects and structures online jobs, matches them to a profile and helps find relevant job for a job seeker.

This study is different from that of Text Kernel (2016) in that it tries to generate a common terms database to represent job descriptions and vacancies in same cluster to maximize the likelihood of their appearance in the suggestion list. It does not only use active vacancies and user profiles as in Text Kernel (2016) rather it uses user profiles, active and historical vacancies, standard job descriptions and social network data.

Venn – Targeting job seekers in Singapore, Venn (Vennjobs, 2016), executes algorithms that collects job seeker information via a web survey and matches it with employer. Venn mainly focuses more on cultural fitness than the skills.

Table 2.3 provides the summary of online job matching systems selected based on their feature similarity to this study according to the criteria set to measure quality of information system in Eppler (2001, 2006).

Reference	**Summary**	**Limitations Identified**
Beansprock (2017)	allows active and casual job seeker to provide their preferences, mainly location and organizational culture, and skills to send the matching jobs by email.	provides limited choice when collecting job seeker preferences and skills. It only works for IT jobs in the US
Irishjobs (2017)	IrishJobs provides job search features grouping them into job advertisements based on recruitment type, roles and skills, location, employment type and salary to facilitate searching.	It is primarily a job searching (as opposed to matching) system.
StepStone (2016)	collects qualification information and preferences from job seekers who subscribe. It also lets job seeker browse through vacancy categories	does not have personalized recommendation of jobs
Text Kernel (2016)	extracts job seeker profiles from resumés and matches them with vacancies	does not incorporate job seeker preferences
Vennjobs (2016)	collects job seeker information via web survey with focus on looking for skills and cultural fitness	does not include social network information about job seekers

TABLE 2.3: Review of existing web-based job matching systems

All these systems with the exception of Beansprock use either close- or open-ended user interface elements to collect data from job seekers. That is, they do not provide assistance in data entry, which will then have a drastic effect on data quality due to dropout (Belloni et al., 2016). Moreover, they focus on collecting information about working culture, working hours and other non-skill related data from job seeker. The objectives seem to collect job seeker preferences in order to match to vacancies.

Unlike matching as "key to performance" explained in Target Training International (2013), this study does not include cultural and behavioral factors affecting matching. It also does not include the effect of job mismatches such as the cost of disengagement of employees and the consequence thereof in the team performance.

Limitations in the literature motivated this research to proceed in applying data-intensive approach on unstructured natural language text, which is grossly ignored in skill research. The originality of the solution lies in the fact that matching job seeker to vacancy used the entire content (as opposed to only job titles) of resumé and vacancy data. This research added perspectives to job seeker and vacancies by incorporating social networking and occupational standard data, respectively, with the prime objective of improving the relevance of the job seeker and vacancy data.

Chapter 3

Theoretical and Conceptual Foundation

This chapter presents a theoretical and conceptual framework – explaining various concepts – employed in the course of this research with overview of different techniques utilized for data collection, knowledge extraction, representation, modeling, prediction and presentation of job recommendations in accordance with human computer interaction and job matching in order to provide conceptual and theoretical foundations on which this research is based. It discusses the concepts in job matching as well as mismatch and the associated societal consequences such as unemployment and underemployment. It also explored the machine learning techniques available with special emphasis on deep learning that is used in the job matching processes in this research.

Conceptual framework of this research is developed based on reviews of literatures and existing systems (cf. Chapter 2). In order to address the lack identified in literatures and existing systems, data collection methods that employ web mining are devised. Natural language techniques that analyze job seeker and vacancy data to make it suitable for further processing are explored. Machine learning techniques for text analysis were reviewed in order to decide the selection of appropriate method for this research.

The framework depicts how the data web mining methods feed the collected data to natural language processing components. It also demonstrated the deep feature learning of vacancy and job seeker followed by the matching and recommendation. More specifically, the conceptual framework presents a job matching system that comprises three principal components – i) job seeker modeling, ii) vacancy modeling and iii) matching. These three components are discussed in three separate chapters: job seeker

analysis and modeling, vacancy analysis and modeling and job seeker to vacancy matching in Chapters 4, 5, and 6, respectively.

3.1 Job Matching

Job matching is the process of assigning the job seeker that fulfills the requirements of a job. Greenberg (2010) defined job matching as "the process of matching the right person to the right job based upon the individual's inherent motivational strengths" that requires thorough understanding of the vacancy and the job seeker. It is important to study job matching in order to address adverse phenomena that occur in job market such as unemployment, underemployment and unfilled job vacancies, each of which cause further undesired economic and social consequences.

Unemployment, underemployment and unfilled job vacancies happen due to industrial restructuring, business process reengineering, downsizing and skill mismatch (Callan and Bowman, 2015, Greenwood, 1999, McKee-Ryan and Harvey, 2011, Şahin et al., 2014b). International Labour Organization (ILO) defined underemployed using time-based definition which states individuals who were willing and available to work additional hours, during the reference week, but worked fewer than a certain number of hours (Greenwood, 1999). These minimum number of hours used as a threshold to determine underemployment vary from country to country. Unemployment, on the other hand, refers to "persons who during the reference week did not work nor had a job but who were willing to work (they show that they are willing to work by actively seeking work) and were available to work" (Greenwood, 1999). Both underemployment and unemployment are caused by skill mismatches that can be addressed through proper job matching. Manifestations of these challenges, in turn, have adverse effects in the society, i.e., unfilled vacancies and unemployment increase pressure causing economic instability while underemployment has adverse effect on the workers. Reynolds and Myers (2012) found out that there is a strong evidence that shows the adverse effects underemployment causes to workers, namely psychological distress, poor health condition, consequent lower wages, and slow career progress. This corroborates the urge to applying efforts to alleviate skill mismatch.

Skill mismatch is the gap between available workers with requisite skill for the job they hold (Şahin et al., 2014b). Different disciplines have varied perspectives of skill mismatch. It can be quantitative or qualitative. Quantitative mismatch is the discrepancy between labor supply and demand. Labor supply

refers to the number of available skilled human resource in the labor force, whereas demand refers to the number of available job vacancies requiring skills in the job market. Qualitative mismatch is the other perspective regarding skill mismatch. It occurs when workers are assigned to a job for which they do not have the skills to perform it effectively. From here on, the term skill mismatch (or job mismatch) is used to refer to qualitative mismatch.

Employment mismatch between job seeker and vacancies can be quantitative or qualitative. In the first case, it is a discrepancy between the available number of jobs and the number of people possessing the required skills and qualifications for those jobs.

In the second case, however, qualitative mismatch refers to job seekers landing on job that are not suitable for them because the jobs do not fulfill their preferences, or they do not have the requisite skills, i.e., technical skills and soft skills. Qualitative skill mismatch has two forms: i) over-qualification or skills under-utilization (Kazan, 2012) and ii) under-qualification or under-skilled (Quintini, 2011).

Over-qualification often occurs if persons find themselves in a work situation where their skills and qualifications are too advanced to be applied for their job (Kazan, 2012). As a result, the worker's full productive capacity is not utilized as it should. This can happen when individuals have no choice except getting employment with a level below their qualification, skill level and/or experience. Under-qualification, on the other hand, is when an individual in a work situation does not have sufficient skills to perform on the job (Quintini, 2011).

Skill mismatch occurs due to either i) skill deficit on the part of the worker as a result of changes in requirements of the job, e.g., change in technology, ii) changing skill demand in the job market, for example, due to emergence of new jobs with new skill requirements, or iii) incorrect matching of job seeker to vacancy, for example, recruitment of a job seeker that does not have the requisite skills for the job. With data surge as a result of fast-changing technology, recruiters are often overwhelmed by the task of analyzing job seeker profiles and vacancy requirements to perform job matching process and end up with incorrect matching.

Job mismatch as one cause of underemployment contributes to adverse effects on workers. Beukes et al. (2016) identified three job mismatches that cause underemployment: i) demand mismatch (i.e., skills gap between existing education and emergent job needs), ii) education supply (i.e., producing fewer graduates than a field requires), and iii) qualification-job (i.e., caused due to people moving into fields different from what they have studied). While all these causes are directly related to either job seeker or job supply, the

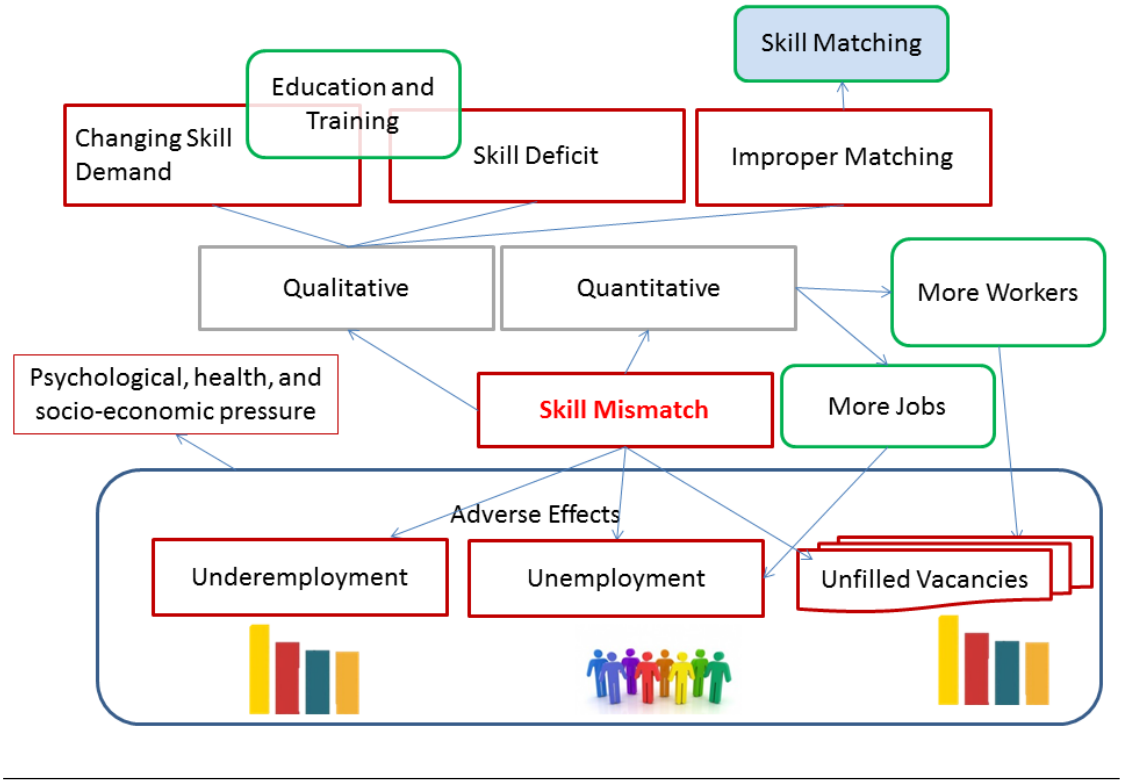


FIGURE 3.1: Conceptual foundation of job matching

problem of correlating job seekers to vacancies should also be given due attention. In order to address the skill mismatches caused by limited labor supply, on one hand, academic institutions provide education and training on skills needed in the job market. On the other hand, policies are crafted to increase jobs in order to address unemployment challenge caused by increased job demand. The labor or job supply is the mere increase of the number of job seekers or jobs, respectively, and has little to do with whether the profiles of job seeker fulfill the requirements of the job and vice versa.

The motivation of researching job matching is due to the prevalence of skill mismatch, i.e., on one hand, there is soaring unemployment and underemployment, and on the other hand, employers struggle with unfilled vacancies. As one way of addressing skill mismatch, and therefore its consequences, this study investigates development of automatic job matching using data-intensive methods. The relationship between the challenges in job seeker to vacancy matching, their causes and adverse consequences as well as methods to address them are depicted in Figure 3.1.

The assumption underlying this research is that by analyzing data from various sources, it is possible to understand factors that affect job matching and devise ways to improve the matching process. Collecting

online data about job seekers and vacancies through web mining, improving user experience in data collection from job seeker, supplementing job seeker profile with social networking data and enhancing content of vacancies through occupational standards data help achieve a reasonable job seeker to vacancy matching.

Hence the study applies web mining, NLP and machine learning, focusing on matching skills and job titles of job seekers to vacancies and vice versa. Not accounting for other aspects of employment such as performance appraisal, this study focuses on analysis of text data in order to build a model against which a newly produced vacancy or a new job seeker is compared with for similarity. In other words, this research is aimed at addressing the issues in qualitative mismatches (as compared to quantitative mismatches).

3.2 Enriching Vacancies with Occupational Standards

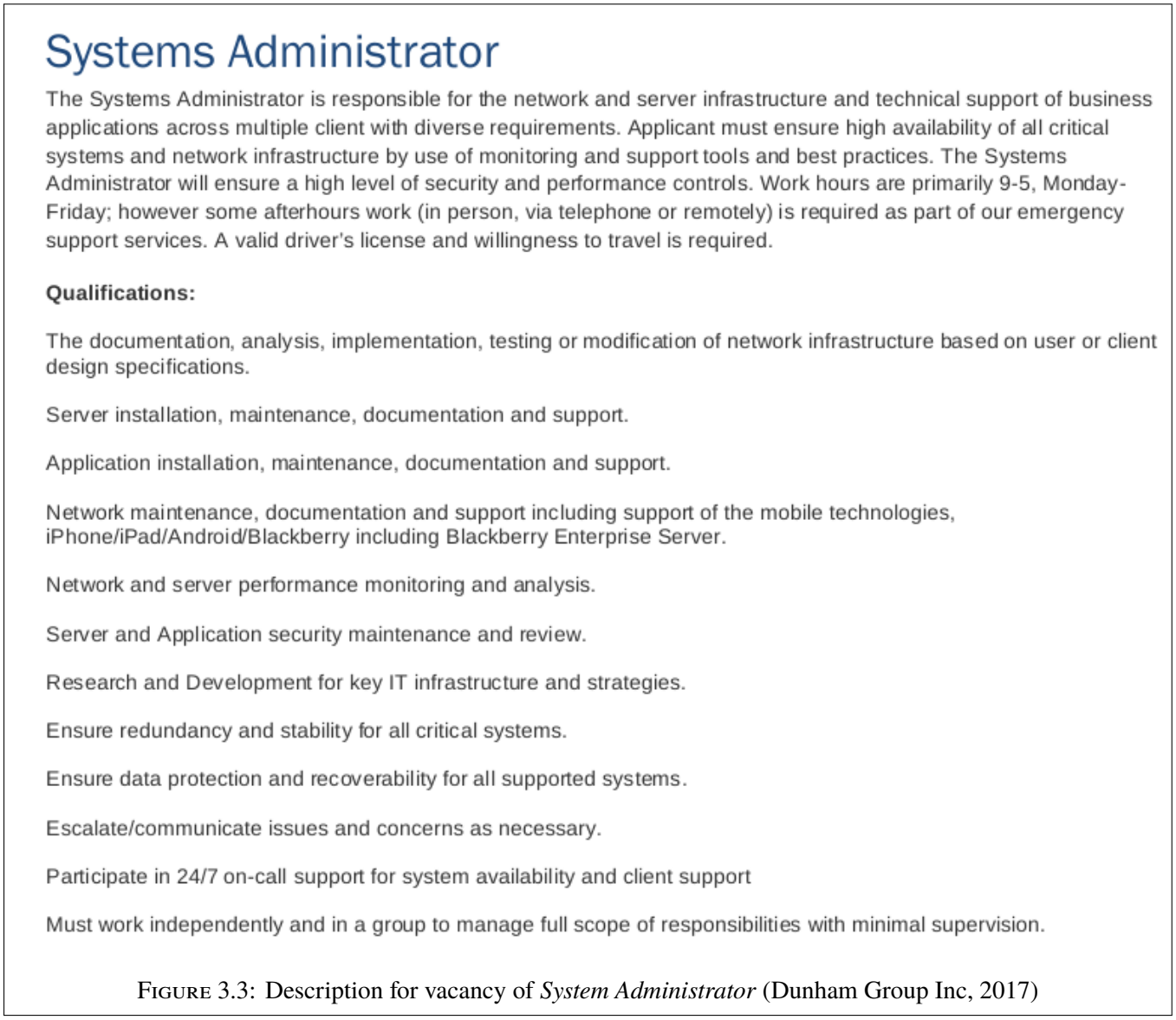
Occupational standards are defined as specifications of the job titles, their descriptions together with associated skills requisite for the job title. Occupational standards are useful coordination mechanisms for improving the match between job seekers and vacancies through specifying the skill and qualification requirements for an occupation. It can be used as a useful guideline for job seekers to developing their careers towards the expectations of the occupation they aim to achieve. Systematic occupation descriptions of these standards are "expected to simplify keeping qualifications up to date and relevant to the needs of the labour market while providing information to learners on the job profile targeted by the qualification." Cedefop (2009)

Often times, countries follow different approaches to developing occupational standards. Cedefop (2009) grouped them into three: i) those that "take the form of or a more or less elaborate but comprehensive classification system providing categories for monitoring the labor market", ii) those "designed as benchmarks for measuring occupational performance" and iii) those that "describe the occupation targeted by a qualification."

Job vacancies that lack information on job details, i.e., descriptions and skill requirements result in job mismatch (Şahin et al., 2014b). Occupational standards contain useful information that can address job mismatch that occurs due to job vacancies' lack of information. Because occupational standards contain comprehensive information about an occupation for which the vacancy is advertised.

FIGURE 3.2: Occupational standard description of *System Administrator* (European Commission, 2017)

This lack can be filled by the data from the occupational standard for the specified occupational title. For example, Figures 3.2 and 3.3 present how the job title *Systems Administrator* presented in occupational standard and vacancy advertisement, respectively, to showcase the discrepancy between content of occupational standards and job vacancy for a job title. From the figures, one can observe that the vacancy advertisement shown on Figure 3.3 is missing important terms: *'access', 'administering', 'backups' 'computer', 'computing', 'configurations', 'control', 'coordinate', 'determining', 'develop', 'diagnosing', 'disaster', 'environments', 'hardware', 'improve', 'information', 'maintain', 'maintaining', 'master', 'monitor', 'networks', 'operations', 'recovery', 'software', 'technology'* which are present in the standard occupation shown in Figure 3.2 for this title. This corroborates the importance of using occupational standards data to enrich vacancies and thereby achieve better job matching.



Systems Administrator

The Systems Administrator is responsible for the network and server infrastructure and technical support of business applications across multiple client with diverse requirements. Applicant must ensure high availability of all critical systems and network infrastructure by use of monitoring and support tools and best practices. The Systems Administrator will ensure a high level of security and performance controls. Work hours are primarily 9-5, Monday-Friday; however some afterhours work (in person, via telephone or remotely) is required as part of our emergency support services. A valid driver's license and willingness to travel is required.

Qualifications:

The documentation, analysis, implementation, testing or modification of network infrastructure based on user or client design specifications.

Server installation, maintenance, documentation and support.

Application installation, maintenance, documentation and support.

Network maintenance, documentation and support including support of the mobile technologies, iPhone/iPad/Android/Blackberry including Blackberry Enterprise Server.

Network and server performance monitoring and analysis.

Server and Application security maintenance and review.

Research and Development for key IT infrastructure and strategies.

Ensure redundancy and stability for all critical systems.

Ensure data protection and recoverability for all supported systems.

Escalate/communicate issues and concerns as necessary.

Participate in 24/7 on-call support for system availability and client support

Must work independently and in a group to manage full scope of responsibilities with minimal supervision.

FIGURE 3.3: Description for vacancy of *System Administrator* (Dunham Group Inc, 2017)

3.3 Knowledge Based Methods in Job Matching

A knowledge base is an organized repository of knowledge in a computer system that consists of concepts, data, rules, and other specifications (BusinessDictionary.com, 2017). Knowledge Management is the explicit and systematic management of knowledge - and its associated processes of creation, organization, dissemination and utilization. It attempts to synthesis Information and Communication Technology (ICT) and some aspects of Human Resource Management (HRM) for acquisition of knowledge (i.e., from individuals and organizations both internally and externally) for use in later stages (Bektas, 2013).

The need for Knowledge Management (KM) is increasing due to increasing globalization of businesses, shift to knowledge economy, expansion of ICT, i.e., computers, smart phones, tablets, cloud computing systems (application and storage anywhere), and emergence of K-workers. Transformation of information is done by human through: comparison, consequences, connections, conversation. For example, knowledge is driven when a person compares some information with previous situation (Baskerville and Dulipovici, 2015). Knowledge can be extracted from humans, i.e., from minds of "knowers" or from organizations, i.e., documents, repositories, organizational routines, processes, practices, norms, and so on. Strategies of KM for conversions of raw information to knowledge include generation of knowledge, identification, capturing, sharing, and exploitation

A number of tools and technologies are available for use to extract knowledge from human such as Enterprise 2.0 (Back and Koch, 2011), which is a keyword-based system. More specific systems that capture recruitment knowledge from human are those developed to collect job seeker information and employers assessment systems.

Knowledge based systems are collection of algorithms that utilize methods to solve problems by i) gathering information through data entry, scanning images, voice inputs or extracting information from various sources, ii) representing and organizing the gathered information through cataloging, indexing, sorting, filtering and linking, iii) refining through mining, projecting, contextualizing and compacting, and iv) presenting and disseminating through pushing recommendation, sharing and alerting.

Various applications of knowledge discovery techniques allow systematic analysis of data and find out implicit relationships among the instances in large data set. However, due to the size and diversity of the data, applying knowledge discovery techniques poses challenge in terms of processing and storage. One solution to overcoming the processing challenge is performing Big Data analysis over distributed platform using Cloud Computing technologies.

Knowledge representation is achieved in any of the following four ways (Turban et al., 2005): Production Rules, Semantic Networks (ontologies), Frames and Formal Logic. Ontology is advantageous in that it is easy to track associations, it is flexible, and it will give us a unified representation that provides “machine-processable semantics of data and knowledge sources” and “facilitates semantic integration, knowledge sharing and reuse” (Maier, 2007).

In data-intensive systems, however, knowledge representation is achieved through models built from patterns discovered through analysis of a large dataset (Wu et al., 2015). This is achieved via machine

learning, i.e., a set of algorithms that perform computation on large data to discover patterns and derive rules. It is also called mining of the data for patterns that match with the requirement (Mohri et al., 2012).

From these knowledge representation techniques, this research uses a data-intensive knowledge representation using lexicons as features (Markman, 2013) extracted from vacancies and job seekers data. The reason for using lexical knowledge representation (Pustejovsky and Boguraev, 1993) technique is partly because the study focuses on extracting patterns from natural language data, as opposed to formulating rules, and partly because the data used in the research is semi-structured, i.e., both the vacancy and job seeker data have structured attributes as well as unstructured text, as discussed in Chapters 4 and 5.

Studies of data-intensive knowledge based systems, i.e., systems that derive dynamic rules from data, involve mining knowledge from Big Data, i.e., large and complex data, to create value in a number of ways (McKinsey Global Institute, 2014): like traditional Relational Database Management System (RDBMS), they make information transparent and usable at much higher frequency; and they help collect accurate data that expose variability and enhance performance. Unlike RDBMSs, they can help companies to provide precisely tailored products or services by allowing narrow segmentation of customers; they improve decision-making based on sophisticated analytics; and can be used to improve the development of next generation products and services.

In order to analyze big data and generate insights, five key approaches are employed: Discovery tools, Business Intelligence (BI) tools, In-Database Analytics, Hadoop Framework, and Decision Management (Oracle Corporation, 2013). In the context of occupational management, BEKO-SMS (Hiermann and Höfferer, 2003) utilizes knowledge management and knowledge-based systems as applied in Skill Management and Personal Development.

Knowledge extraction in job matching is similar to content based recommender systems (Lops et al., 2011) in that it finds matching patterns for suggestion. Among existing systems that use recommender system to help users pose their queries include: Informed Recommender (Aciar et al., 2007), News@hand (Cantador et al., 2008) and (Khobreh et al., 2013). Informed Recommender uses consumer product reviews to make recommendations. This system converts consumers' opinions into a structured form by using a translation Ontology, which is exploited as a form of knowledge representation and sharing. News@hand (Cantador et al., 2008), on the other hand, is a system that adopts an ontology-based representation of item features and user preferences to recommend news. Khobreh et al. (2013) applied ontology in designing

recommender system for learning material to health professionals. In data-intensive systems, however, knowledge extraction is attained from patterns in the data.

As this research focuses on matching job seekers with vacancies using semi-structured data, it applies data-intensive methods, i.e., machine learning for knowledge extraction (Ayodele, 2010, Mooney and Bunescu, 2005). Prevalence of big data, time consuming task of extracting knowledge from experts, difficulty of access to (and huge cost of) experts to formulate rules which can be used by Artificial Intelligence (AI) systems are reasons for choosing a data-intensive approach – machine learning – to perform matching candidates to job seekers in this research.

3.4 Machine Learning and Natural Language Processing

Machine learning is defined as an area of AI that provides programs the ability to learn from experience (known data), i.e., without being programmed explicitly to improve performance or make prediction based on unknown data (Mohri et al., 2012). It is not always possible to solve complicated problems that deal with unreliable rules to get an output for a given input. Machine learning helps solve problems when we do not know how to write a programs with pre-determined results, i.e., it is the development of adaptive algorithms that can change when exposed to new data.

ML is applied in various problem domains such as NLP, image processing, speech recognition, recommendation and fraud detection to perform classification, regression, clustering and dimensionality reduction (Mohri et al., 2012). Availability of massive computing resources coupled with huge volume of data makes machine learning a viable solution to solve complex problems such as matching.

In machine learning, there are a number of learning scenarios based on the type of data available for training, the order and method of feeding training data to the learner and method of using test data to evaluate the learning algorithm. The most common learning scenarios are: i) supervised learning, ii) unsupervised learning, iii) semi-supervised learning, and iv) reinforcement learning (Mohri et al., 2012).

Supervised learning is learning to predict an output for a given input vector. It is used when labeled training data is available. The learning algorithm is trained with a set of labeled example data so that it makes prediction for all unseen test data. Supervised learning is the most common learning scenario applied in solving classification, regression, and ranking problems. In supervised learning, each training

case consists of an input vector x and a target output t. The algorithm is trained with the vector x and target output t so that it can predict unknown target output from a given input vector. Types of supervised learning include regression and classification. Regression is where the target output is a real number or a whole vector of real numbers, for example, the price of an item in 6 months' time. ii) Classification is when the target output is a class label, e.g., the simplest case is a choice between 0 and 1. In classification, we can also have multiple alternative labels (Bhavsar and Ganatra, 2012, Mohri et al., 2012).

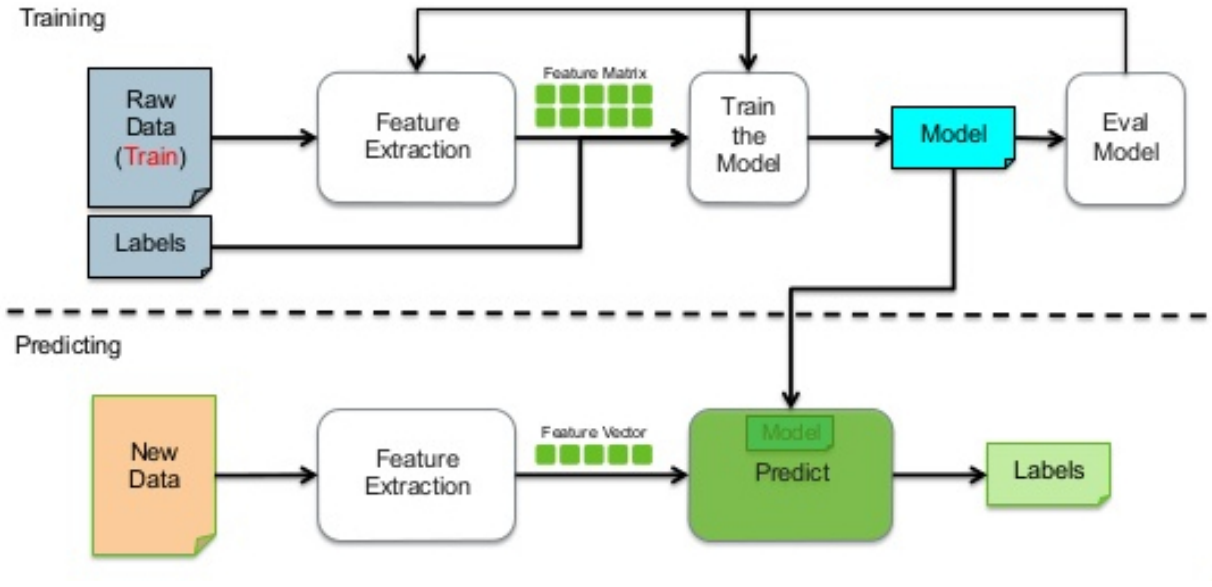


FIGURE 3.4: Workflow in supervised learning (adapted from: Hortonworks Inc. (2014))

Unsupervised learning is defined as "a type of machine learning algorithm used to draw inferences from datasets consisting of input data without labeled responses." (MathWorks, 2017) Unlike supervised learning, the input data for supervised learning do not have target attribute. The algorithms explore the data to find some intrinsic structures in them. That is, unsupervised learning studies how algorithms learn to represent input patterns in a way that reflects the statistical structure of the overall collection of patterns in the data (Dayan et al., 1999). Unsupervised learning is used when large unlabeled training data is available. The learning algorithm receives large unlabeled training data, develops patterns and makes predictions for unseen test data. Unsupervised learning is more appropriate at discovering a good internal representation of patterns in the input data. Unsupervised learning is applied in solving clustering and dimensionality reduction problems (Coates et al., 2011, Mohri et al., 2012). When exposed to large data, unsupervised learning algorithms discover similarities within the features of the data and cluster the data

that have similar patterns together. Unsupervised learning algorithms are also useful for dimensionality reduction (Dash et al., 1997), i.e., selecting only relevant features that help solve the problems. By doing so, the algorithms reduce the size of features that will be considered for analysis, thereby reducing the computational resource requirement of the solution.

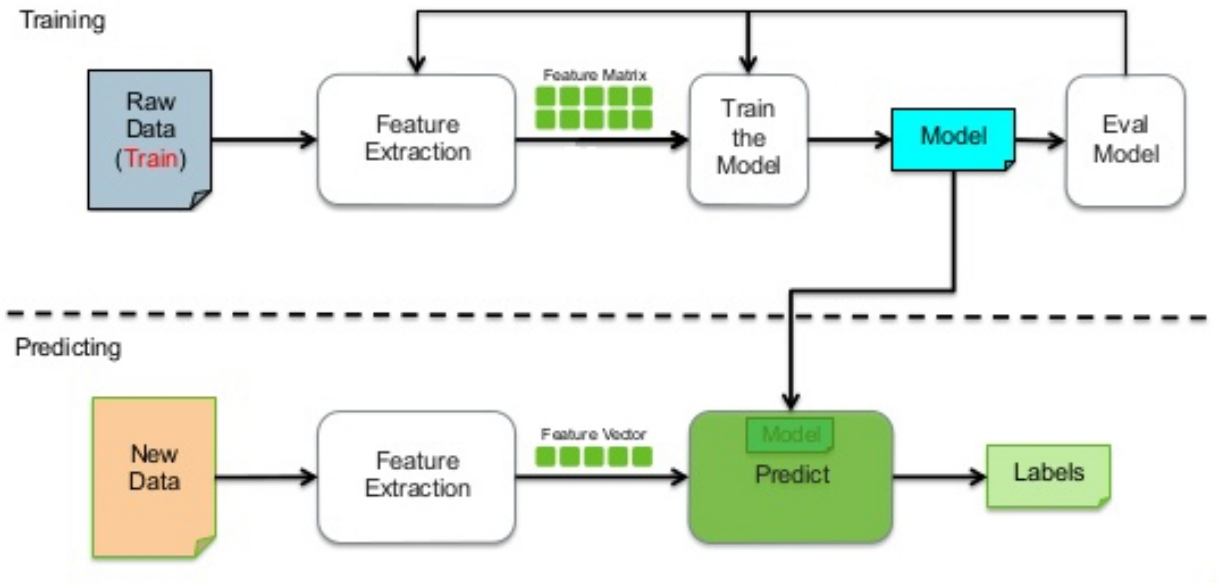


FIGURE 3.5: Workflow in unsupervised learning (adapted from: Hortonworks Inc. (2014))

Semi-supervised learning is suitable when we have both labeled and unlabeled data. The learning algorithm takes a training data that consisting of both labeled and unlabeled data, and produces predictions for unseen input data. Unlike supervised learning, with semi-supervised learning, the cost of labeling data can be reduced by using only getting partial training data labeled. It is applied in solving problems of classification, regression and ranking (Mohri et al., 2012).

Reinforcement learning is when training and testing phases are intermixed. The learner has continuous and active interaction with the environment where it is rewarded for correct predictions. It is learning aimed at selecting an action that maximizes payoff (Mohri et al., 2012, Szepesvári, 2010).

On grounds of lack of labeled data and limited resource to get the data labeled by experts, this study opted to utilize the advantage that unsupervised learning provides in learning from unlabeled data. There are a number of unsupervised learning algorithms, i.e., algorithms that learn from unlabeled data (Mohri et al., 2012). Artificial Neural Network (ANN) is the most flexible and efficient for variable input size

(Ayodele, 2010, Celebi and Aydin, 2016, Längkvist et al., 2014). The job seeker and vacancy data is textual data with variable document size and dimension. As a result of this, ANN is the most logical machine learning algorithm to apply in this research. ANN is a network of interconnected nodes that simulate the interconnection of neurons in the brain. As described by Schmidhuber (2015), ANN is composed of

> *many simple, connected processors called neurons, each producing a sequence of real-valued activations. Input neurons get activated through sensors perceiving the environment, other neurons get activated through weighted connections from previously active neurons.*

Typically, ANN has three layers: i) input layer, ii) hidden layer and iii) output layer as depicted in Figure 3.6. The input layer is first layer which is triggered by input. The output layer is the last layer which produces the output. The hidden layer is the layer between input layer and the output layer.

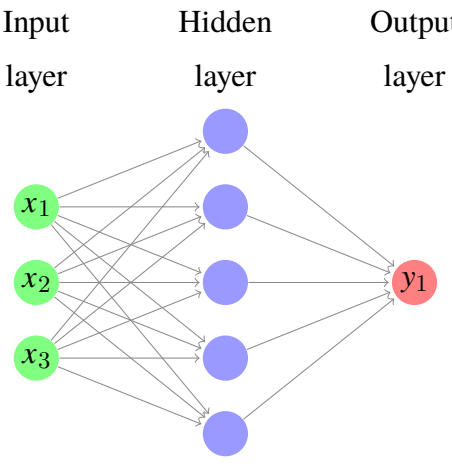


FIGURE 3.6: Artificial neural network

The input layer accepts input from external sources, performs computation and sends its results to hidden layer. The hidden layer, on its part, processes the input data that it accepts from input layer, and sends the result to output layer. The output layer processes the data it accepts from input layer, processes it and sends the result to the external world when the condition on the activation function (Schmidhuber, 2015) is satisfied.

ANN can have multiple hidden layers, in which case it is called Deep Neural Networks (DNN). Deep learning is learning that involves ANN that has multiple hidden layers (Schmidhuber, 2015). In ANN,

there are several unsupervised learning methods that utilize deep learning model architectures and learning algorithms such as feed forward neural networks, recurrent neural network, multi-layer perceptrons, and convolutional neural networks (Längkvist et al., 2014).

Feed forward ANNs are the most common type of neural network in real world applications and can, in theory, compute any function. However feed forward ANNs with a single hidden layer, i.e., learning by shallow architectures, is not so efficient as compared to DNN (Bengio et al., 2009, Deng et al., 2014).

3.5 Deep Learning with Convolutional Neural Networks (CNN)

DNN is when there are more than one hidden layers, and hence "deep" neural networks, for example, a deep neural network with four hidden layer is shown in Figure 3.7. Recently, machine learning has achieved big success with DNN in various applications (Deng et al., 2014) such as image matching, speech recognition, and NLP by displaying significant gains over state-of-the-art shallow learning methods. DNNs have shown promising results by outperforming other machine learning techniques in NLP (Yan et al., 2016). There are different implementations of DNN, CNN being the most popular and flexible neural network that proved effective in image processing. CNN is chosen for this research on the basis of its effectiveness and flexibility. It also enables us to represent text features using terms (cf. Section 3.6) in a similar manner we represent images using pixels. Because this research utilizes unlabeled data, unsupervised learning approach is employed using deep learning CNN.

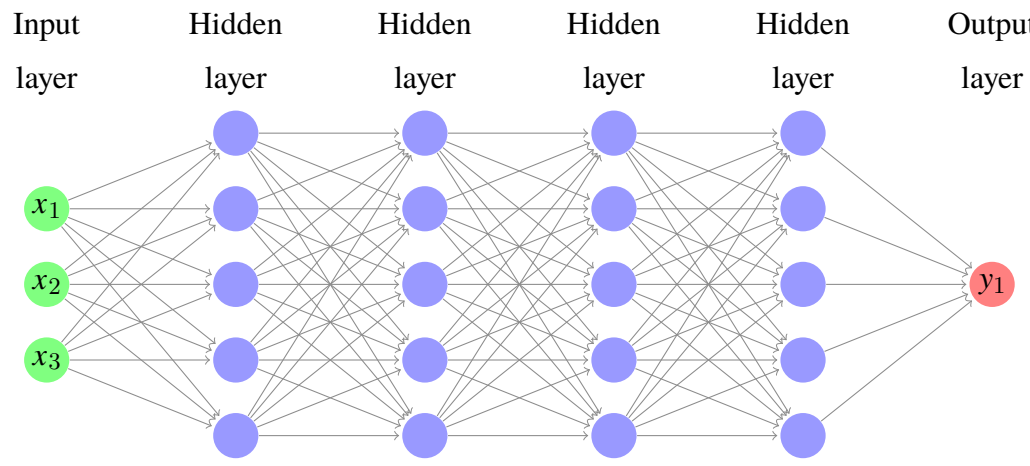


FIGURE 3.7: Deep artificial neural network

For example, deep learning for semantic matching of short text through re-ranking achieved a better result as compared to other methods (Severyn and Moschitti, 2015). Severyn and Moschitti (2015) implemented semantic matching of text using deep learning architecture shown in Figure 3.8.

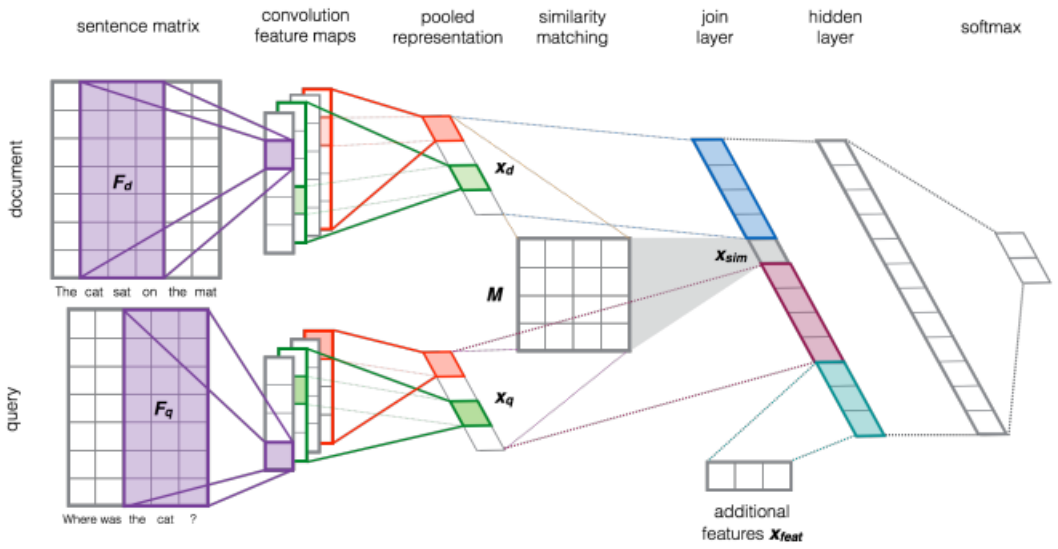


FIGURE 3.8: Deep artificial neural network using CNN (Adapted from Severyn and Moschitti (2015))

For this study, which involves analysis of natural language text in job matching, a deep learning with CNN (Dumoulin and Visin, 2016) is used. The choice of deep learning CNN is because deep learning neural network architectures perform better on learning patterns in complex data than the traditional neural networks as they have more hidden layers (Bengio et al., 2009, Deng et al., 2014). Yet, another reason is that deep learning networks can be trained for unsupervised and supervised learning tasks (Bengio et al., 2009).

3.6 NLP for Data-intensive Job Matching

Working with unstructured text of vacancy and job seeker document utilizes a document vector in order to represent them using statistically most important words contained in the document. This presupposes feeding the features to the unsupervised feature learner which will then be used to select features for similarity analysis. The features of the document vectors will be term weights, i.e., terms/attributes that

exist in the documents. Term weighting is evaluating the importance of a term in representing a document, i.e., the more important the term is, the more it represents the document.

Term weighting is done using different methods (Jurafsky and Martin, 2009). The most common ones are Term Frequency (TF), Inverse Document Frequency (IDF) and their combination TF*IDF (Jurafsky and Martin, 2009). The assumption with TF to use it for term weighting is that the more often a term appears in a document, the more important the term is, thus term frequency, as shown in Equation 3.1.

$$tf_{t,d} = \sum_{t' \in d} f_t(t') \text{ where } f_t(t') = \begin{cases} 1 & \text{if } t' = t \\ 0 & \text{otherwise} \end{cases} \tag{3.1}$$

For example, consider the following four documents extracted from job seeker and vacancy dataset, d_1 = <*proven experience implementing Microsoft SharePoint integration*>, d_2 = <*Experience with implementing MySQL under high query volume, large data integration*>, d_3 = <*Working at OpenDNS means being surrounded by passionate, intelligent and creative people that are determined to disrupt the Internet security industry with innovative ideas*> and d_4 = <*core competencies include deep business process and industry expertise, advanced analytics and research capabilities, comprehensive IT infrastructure knowledge, and proven ability to implement enterprise solutions*.

The $tf_{sharepoint}$ in d_1 and d_2 is 1 and 0 while the $tf_{integration}$ in d_1 and d_2 is 1 and 1 respectively. On one hand, the term *sharepoint* has the ability to represent d_1 well and discriminate it from d_2, as the latter does not contain the term. On the other hand, the term *integration* has the ability to represent d_1 and d_2 but cannot discriminate d_1 from d_2 very well, as the term exists in both documents.

Although the frequency of occurrence of a term describe a document well, it is not enough to discriminate it from other documents. As a result, term frequency is not sufficient for measure of relevance. Because terms that present in too many documents do not help to separate documents even if they have high frequency.

IDF, on the other hand, measures the capacity of a term to discriminate the documents from other documents. Contrary to TF, rare terms have higher IDF and thus better at discriminating documents from each other. In order to compute the IDF in the document collection D, we take the total number of documents, i.e., |D|, and divide it to the number of documents containing the term. Then we take the logarithm of the result. In other words, IDF of a term is the logarithm of the ratio of the total number of

documents to the number of documents containing the term as shown in Equation 3.2.

$$idf(t, D) = \begin{cases} log(\frac{N}{N_t}), & \text{if } N_t > 0 \\ 0, & \text{otherwise} \end{cases} \tag{3.2}$$

where: N = |D|, i.e., total number of documents

N_t = number of documents containing term t

From the above example, the idf(*sharepoint*) and idf(*experience*) in the given document collection is 0.6 and 0.3, respectively.

Combining TF and IDF to measure importance of the term as determined by its frequency and its capacity to distinguish between documents, the weight of a term is computed using Equation 3.3.

$$w(t, d, D) = tf_{t,d} \times idf_{t,D} \tag{3.3}$$

where: $tf_{t,d}$ = frequency of the term t in document d

$idf_{t,D}$ = idf of term t in D

Terms that occur many times in few documents have the highest TF*IDF followed by terms that occur few times in few documents. Terms that occur in too many of the documents will have the lowest representation and distinguishing capacity. In this study, term weighting is done using the combined metrics, i.e., TF*IDF because this metric combination balances the representation and discrimination capacity of a term (Domeniconi et al., 2015). Thus, the weight of a term in a particular job seeker document, is the product of the *tf* of the term in the document multiplied by the *idf* of the term in the job seeker document collection. Likewise, the weight of a term in a particular vacancy document, is the product of the *tf* of the term in the document multiplied by the *idf* of the term in the vacancy document collection.

As, in the above example , if*idf of term *sharepoint* in d_1 is 0.6 and that of *integration* in d_1 is 0.0. Likewise, if*idf of term *sharepoint* in d_2 is 0.0 and that of *integration* in d_2 is also 0.3. Thus, TF-IDF

method heavily penalises the term *integration* as it occurs in all documents in the collection. This makes *sharepoint* an important term for d_1 from the context of the given document collection.

Having the vacancy and job seeker documents represented with feature vectors of term weights, we can measure and rank their similarities in order to suggest matching ones. There are a number of techniques to measure similarities between objects (e.g., vacancies and job seekers in this case) having feature vectors (Yang and Watada, 2011). Once the documents are represented using the term weights, computing the similarities between them is the next task. Among the term-based similarity measures used for comparing the similarity of text in natural language processing are: Euclidean distance, Dice's coefficient, Jaccard similarity, and cosine similarity (Gomaa and Fahmy, 2013).

Matching job seeker and vacancies documents largely depends on measuring the semantic similarity between documents. Latent Semantic Analysis (Jurafsky and Martin, 2009), a corpus-based similarity measure (Gomaa and Fahmy, 2013), is the most widely used and useful technique in representing and analysis of documents in NLP.

Document-document similarity is common in search engines to match queries and documents that are mapped in a low-dimensional space. In the last couple of years, researchers have applied CNN to different NLP studies and reported significant successes, e.g., Dos Santos and Gatti (2014) utilized CNN for semantic analysis of text. In job matching, representing job seeker and vacancies pose several specific challenges. For example, there are many closely similar job titles, job specifications set by several employers are unique, and there is a high correlation between the features. Matching candidate job seeker to vacancy based on the job description and/or titles is not sufficient especially when features such as vacancy expiry date are not taken into account, in which case an expired job vacancy may be matched. Hence, very little has been done in the field of job matching using the whole natural language text. This research applies deep learning using CNN (cf. Section 3.5) and other NLP techniques for job seeker to vacancy matching according to the conception of bidirectional job matching framework discussed in Section 3.7.

3.7 Conception of Bidirectional Job Matching

Based on the theoretical and conceptual foundations discussed, the conception of the system studied in this research depicting the components as well as methods is presented in this section. Derived from the gaps

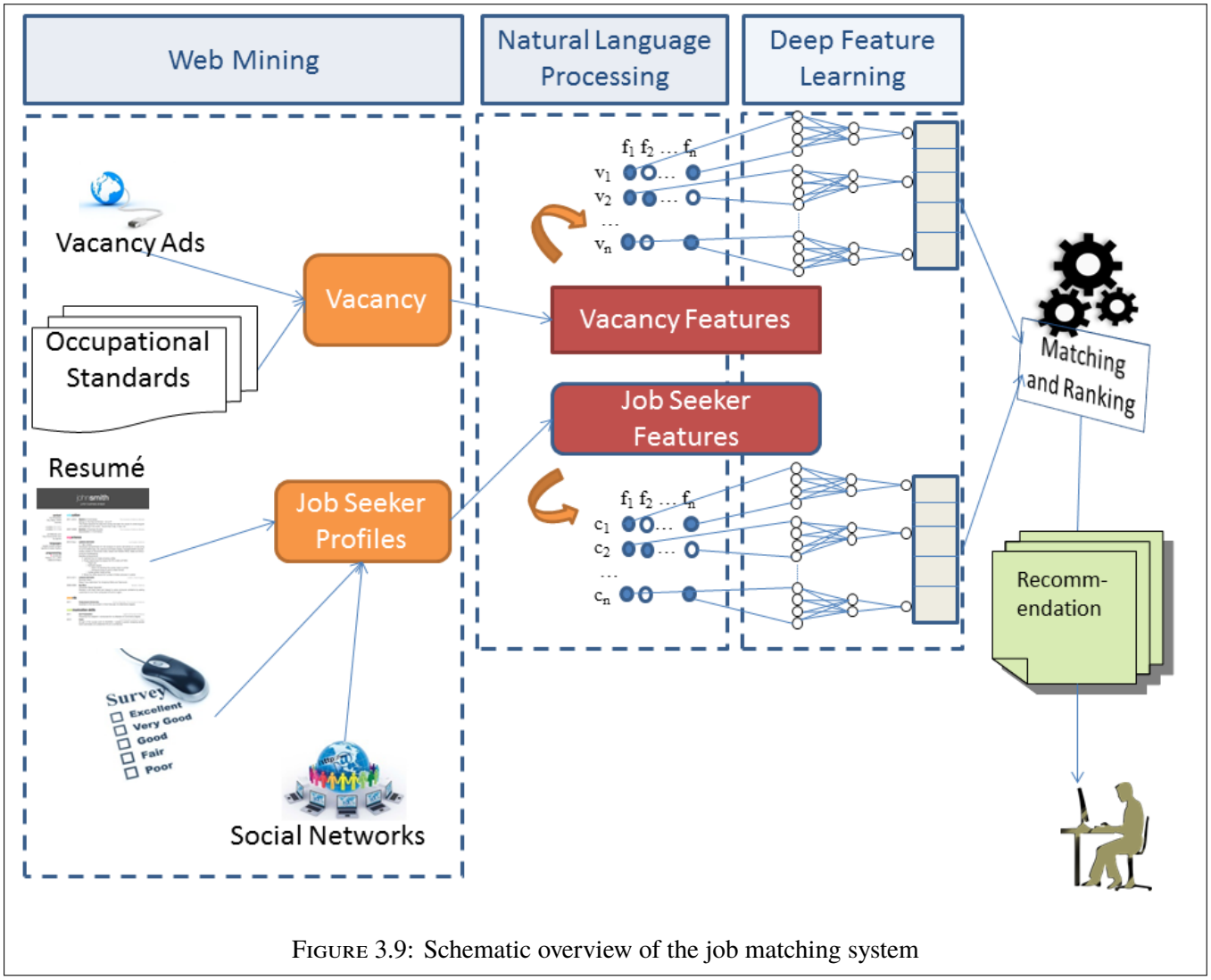


FIGURE 3.9: Schematic overview of the job matching system

in the related literature, theoretical and conceptual foundation, the general framework of the job matching system composed of three principal components is developed, namely i) job seeker modeling, ii) vacancy modeling and iii) matching. Figure 3.9 depicts the overall data and process flow in the framework. It shows collection, processing, and presentation of the job seeker and vacancy data. Job seeker data is composed of data from online resumes, web survey data and social networking data. Likewise, vacancy data is composed of vacancy advertisement enriched with occupational standard data. Natural language processing is applied on the collected job seeker and vacancy data to represent them for further processing. Then deep learning algorithms are applied for feature selection that are then used to match job seeker with vacancies. Finally, recommendations are generated to the user interface. Job seeker data collection, analysis and modeling, which involves the development of a model of job seeker representation based on inputs from web survey data collection and analysis; online resumé data collection, extraction and

analysis; and social network data collection and analysis is discussed in Chapter 4. The source of the data, the procedures of collection and analysis of the job seeker data, and the development of the job seeker models are discussed in subsequent sections in the chapter.

Vacancy analysis and modeling, on the other hand, deals with the collection of vacancy data from open Internet sources via web crawling and web mining. It also includes tasks for extraction of data relevant for representation of the vacancy, enriching the vacancy data with data from occupational standards, and building the vacancy model. Vacancy modeling together with the details of data sources and their structures, methods of collection, extraction and analysis is covered in Chapter 5.

Matching, which is the ultimate goal of this research, is yet another principal component that involves selection of relevant job seekers for a given vacancy, on one hand, and selection of relevant job vacancy for a given candidate job seeker, on the other hand. It selects, filters and ranks job seekers as well as vacancies based on the job seeker and vacancy models. Approaches and algorithms utilized to perform matching job seekers to vacancies are discussed in detail in Chapter 6.

Chapter 4

Job Seeker Analysis and Modeling

Matching job seeker to vacancy involves decision to whether or not a job vacancy is relevant, given the profile of job seeker and vice versa. There are a number of problems associated with deciding relevance in job seeker to vacancy matching. The first is lack of information about job seeker due to job seekers inability or resistance to provide sufficient data about themselves; the second is difficulty in modeling job seekers and vacancies, and the third is complexity of matching process itself. Two fundamental components are required in order to decide a given job vacancy is relevant to a job seeker: a) an understanding and modeling of the job seeker and b) an understanding and modeling of the job vacancy. The former is discussed in this chapter while Chapter 5 covers the latter.

Understanding and modeling job seeker using data about the job seeker, requires collecting as much data and learning patterns in the data. This chapter focuses on exploring the data and methods necessary to analyze and understand a job seeker, and provides a comprehensive job seeker model as an input for job matching. Prior to matching job seeker with vacancy, the first step is to collect job seeker data, i.e., job seeker resumé, web survey and social networking data that can describe job seeker in good detail. The purpose for collecting more information about job seeker than merely using resumés is to get better matching. While resumés tell a lot about the job seeker, social network and survey data add a perspective on top of it, i.e., from social networking data, one can get how people see the job seeker.

The collected data needs cleaning in order to remove noisy and inconsistent data. These data from different sources are integrated, i.e., they are combined to form one unified dataset of job seeker. This

dataset is the basis for selecting data that is relevant for analysis. Through utilization of natural language methods, the job seeker data is transformed into candidate job seeker feature matrix.

The motivation of this work is to pursue two major challenges in job seeker-to-vacancy matching. On one hand, jobs are not filled with the right applicant (Belloni et al., 2016), because i) job seekers may not have access to the right jobs due to overwhelming volume of data; ii) vacancies do not have complete information about the description, requirements, expectations and provisions; iii) job seekers do not incorporate complete information in their application. On the other hand, employees quit or get fired after they assume the position because they under- or overstating their suitability to the job during their application (Branham, 2012).

This chapter presents the process of job seeker data analysis and modeling for job matching. It explains the scope, data, methods and techniques used in collecting, preprocessing, integrating, analyzing and modeling of job seekers with the goal of modeling job seeker for automatic job matching.

4.1 Job Seeker Data Collection and Integration

The data used in job seeker analysis includes online resumé of job seekers, survey data collected from job seekers and job holders as well as social networking data. The reason for using these three types of data about job seeker are to get insight about the job seeker from different perspectives: i) from the job seeker himself in the web surveys, ii) from his connections in social network, iii) and from evidences like education and work experience listed on resumé.

4.1.1 Data Source

Job seeker dataset includes resumés, occupation survey and social network data. Job seeker resumés were collected from several online sources using web mining techniques. Web mining is the task of identifying the composite structure, which can be represented as a template that are filled by individual pieces of structured information from the web (Witten and Frank, 2005). Occupational survey data was obtained from Wageindicator Foundation (2016) and social network data from Stack Exchange Inc. (2016). The reasons for choosing Wageindicator Foundation (2016) and Stack Exchange Inc. (2016) as data source

are availability, comprehensiveness and volume of data they provide. That is Wageindicator Foundation (2016), being a partner of Eduworks project (Eduworks, 2014), provided the web survey data and the data of StackExchange (Stack Exchange Inc., 2016) is publicly available. The data obtained from these sources is preprocessed in such a way that it suits the implementation of the system.

4.1.2 Job Seeker Data Collection

Three types of data about the job seeker, i.e., i) resumé data, ii) social network data and iii) survey data, are collected. Resumé data is collected from the web through web mining; job seeker survey data is obtained from project partner Wageindicator Foundation (2016); and social network data is fetched from publicly available dataset of Stack Exchange Inc. (2016). Job seeker resume was collected from web sites that collect resumés from their subscribers. Figure 4.1 shows a sample resume data out of which useful features that describe the job seeker are extracted.


```
Machine learning technologies in the areas of Deep Learning, Neural Nets, Clustering et al <br/>â€¢ Software engineering technologies in the areas of
Services, API&#039;s, Ap NUI&#039;s et al <br/>â€¢ Software engineering practices such as Agile, Devops, CI &amp; CD <br/>â€¢ Experience with data
science tools (R, NumPy), NoSQL databases (Cassandra, MongoDB), data modeling and statistics. <br/>â€¢ Passion for designing algorithms. <br/>â€¢
experience in Service Oriented Architectures, Docker, Puppet, Chef, Ansible. <br/>â€¢ Implementing machine learning algorithms, ideally probabilistic
methods <br/>â€¢ Python and its C API <br/>â€¢ Hands on experience in parallel programming <br/>â€¢ CUDA programming experience <br/>â€¢
Professional software development <br/>â€¢ Building large-scale engineering software</p></div></div><div id="workExperience-Eec7HmKMKpmbah6VyWpLnA"
class="work-experience-section "><div class="data_display"><p class="work_title title">Data Scientist/Data Engineer</p><div class="work_company" ><span
class="bold">Booz Allen Hamilton</span> <div class="separator-hyphen">-</div> <div class="inline-block" ><span >Boston, MA</span></div></div><div
class="separator-hyphen">-</div><p class="work_dates">January 2015 to August 2016</p><p class="work_description">Data Science: <br/>â€¢ through RShiny
programming and NLP I built an interactive Dashboard that shows how the audience reacts to different topics of internet content produced over time by a
commercial client. Other analytics identified the causal <br/>relationships between engagement and different promotional avenues of that content on social
media. Presented work to client&#039;s HQ <br/>â€¢ through python, R scripting and web-scraping I built an econometric model that predicts &#039;import
demand&#039; for all <br/>industries, countries, for the next 5 years. It is being used by a federal agency for workforce allocation. <br/>â€¢
through R scripting and combining various business unit databases I built a machine learning model that predicts <br/>customer behavior. Being used by a
commercial client in the MENA region <br/>â€¢ through python scripting I converted aviation emission datasets from the input format of one environmental
modeling <br/>software to the input format of a new one. The resulting software comparison educated a federal aviation authority to change their
infrastructure <br/>Data Engineering: <br/>â€¢ leveraged my AWS, Hadoop, Kafka and Spark understanding to reorganize the architecture of an <span
class="hl">IOT</span> product to ingest real time data with higher throughput. Product is getting adoption by major US cities <br/>â€¢ through Scala and
bash scripting I built the Spark analytics component of a Kafka-Spark-RShiny real time streaming <br/>pipeline of transportation data <br/>â€¢ built
a Cloudera manager Hadoop cluster on AWS, as well as a Spark Standalone cluster from scratch</p></div></div><div id="workExperience-Eec7HmKMUaqbah6VyWpLnA"
class="work-experience-section "><div class="data_display"><p class="work_title title">Cofounder and Machine Learning Scientist</p><div class="work_company" >
<span class="bold">ImagoRei</span> <div class="separator-hyphen">-</div> <div class="inline-block" ><span >Cambridge, MA</span></div></div><div
class="separator-hyphen">-</div><p class="work_dates">March 2014 to December 2014</p><p class="work_description">Built a mobile application for Android that
takes a photo of a fish and recognizes what species it is. Used machine <br/>learning, computer vision techniques, Matlab, Python, Java</p></div></div>
<div id="workExperience-Eec7HmKMUaubah6VyWpLnA" class="work-experience-section "><div class="data_display"><p class="work_title title">Data Scientist</p><div
class="work_company" ><span class="bold">Int&#039;l Food Policy Research Institute</span></div><div class="separator-hyphen">-</div><p
class="work_dates">September 2013 to January 2014</p><p class="work_description">Measured the causal impact of cash and food transfers on the asset ownership
of Colombian refugees and poor <br/>Ecuadorians. Leveraged experimental econometrics, R, Stata, Excel</p></div></div><div id="workExperience-
Eec7HmKMeLybah6VyWpLnA" class="work-experience-section last"><div class="data_display"><p class="work_title title">Proprietary Trader</p><div
class="work_company" ><span class="bold">Amplify Trading</span> <div class="separator-hyphen">-</div> <div class="inline-block" ><span >London, ON</span></div>
</div><div class="separator-hyphen">-</div><p class="work_dates">January 2011 to September 2011</p><p class="work_description">Traded the E-mini S&amp;P 500,
EUR/USD, Nymex Crude Oil futures contracts</p></div></div></div></div><div class="section-item education-content"><div><div class="section_title">
<h2>Education</h2></div></div><div id="education-items" class="items-container"><div id="education-Eec7HmKMn82bah6VyWpLnA" class="education-section "><div
class="data_display" ><p class="edu_title">CS in Cloud Computing</p><div class="edu_school"><span class="bold">Harvard Extension School</span> <div
class="separator-hyphen">-</div> <div class="inline-block" ><span >Cambridge, MA</span></div></div><p class="edu_dates">June 2014 to December 2014</p></div>
</div><div id="education-Eec7HmKMxt-bah6VyWpLnA" class="education-section "><div class="data_display" ><p class="edu_title">MS in Quantitative Economics and
Econometrics</p><div class="edu_school"><span class="bold">Tufts University</span> <div class="separator-hyphen">-</div> <div class="inline-block" ><span
>Medford, MA</span></div></div><p class="edu_dates">August 2012 to February 2014</p></div></div><div id="education-Eec7HmKMxuGbah6VyWpLnA" class="education-
section last"><div class="data_display" ><p class="edu_title">BSc in Economics and Environmental Policy</p><div class="edu_school"><span class="bold">London
School of Economics</span></div><p class="edu_dates">September 2007 to July 2010</p></div></div></div></div><div class="section-item additionalInfo-content">
<div><div class="section_title"><h2>Additional Information</h2></div></div><div id="additionalinfo-items" class="items-container"><div id="additionalinfo-
section" class="last"><div class="data_display"><p>TECHNICAL SKILLS: <br/> <br/>Machine Learning: Python, NLTK, Scikit-learn, Numpy, Pandas,
Matplotlib and Tensorflow</p></div></div></div></div></div></div>
```


FIGURE 4.1: Sample resume data

When extracted from the raw source, the resume data has structure that forms features categories as shown in Figure 4.2, i.e., *personal information*, *education*, *experience*, *preferences* and skills as well as

their details except for data that affect privacy. Figure 4.2 shows the features taken into consideration for analysis and modeling the job seeker. These features are produced by utilizing feature extraction methods from the data obtained through web mining.

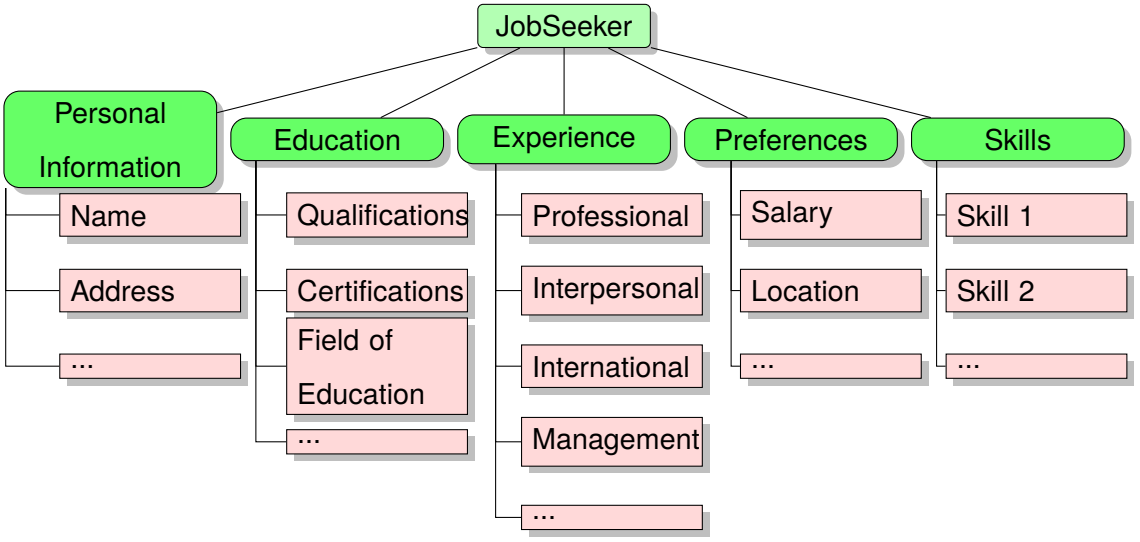


FIGURE 4.2: Features of job seeker from online profiles

In addition to resumé data, the raw social networking data collected for enriching job seeker is obtained in the format shown in Figure 4.3.

<row Id="760" Reputation="113" CreationDate="2013-05-21T00:03:03.000" DisplayName="rick" LastAccessDate="2013-05-29T05:02:14.930" WebsiteUrl="" AboutMe="" Views="0" UpVotes="0" DownVotes="0" AccountId="2793972"/>
<row Id="762" Reputation="6664" CreationDate="2013-05-21T00:15:03.000" DisplayName="John Jensen" LastAccessDate="2016-05-08T01:00:21.343" WebsiteUrl="http://www.linkedin.com/pub/john-jensen/7/571/a24" Location="D.C. metro area" AboutMe="<p>Senior network engineer specializing in automation (Perl, Python, and a little bit of Ruby). I don't care how many certifications or how much experience you have, what company you work for, how old you are, what gender you are, which blogs you write for, and especially what your "reputation level" is. The best engineers are the ones that are on a continual quest to learn and are always asking questions. The minute you stop being the best at what you do is the minute that you assume you are the best at what you do.</p> " Views="14" UpVotes="21" DownVotes="1" Age="33" AccountId="2169507"/>
<row Id="769" Reputation="484" CreationDate="2013-05-21T01:53:48.000" DisplayName="[illegible]" LastAccessDate="2013-07-22T18:01:32.333" WebsiteUrl="http://justnetworked.wordpress.com" Location="Central Ohio" AboutMe="" Views="0" UpVotes="0" DownVotes="0" ProfileImageUrl="https://i.stack.imgur.com/MKudj.jpg" AccountId="2794219"/>
<row Id="774" Reputation="101" CreationDate="2013-05-21T03:18:08.000" DisplayName="World Engineer" LastAccessDate="2015-07-14T10:55:33.627" WebsiteUrl="http://www.linkedin.com/in/westonhopkins/" AboutMe=" Programmers Moderator. Frequently found on the Whiteboard. Concerns or questions? Ask on meta. <h1>SOreadytohelp</h1> " Views="0" UpVotes="0" DownVotes="0" ProfileImageUrl="https://i.stack.imgur.com/aT9pZ.jpg?s=128&g=1" AccountId="407160"/>
<row Id="775" Reputation="20399" CreationDate="2013-05-21T03:52:02.000" DisplayName="[illegible]" LastAccessDate="2016-06-11T12:32:00.650" WebsiteUrl="http://www.pennington.net/" Location="Austin, TX" AboutMe=" Professional packet pusher / network engineer, python & linux addict <ul> My pet project: ciscoconfparse & ciscoconfparse docs. CCIE Emeritus #4483 (R&S) <hr> For personal (non-Stack Exchange correspondence), don't hesitate to email me: mike [at] pennington {dot} net. <hr> " Views="101" UpVotes="150" DownVotes="68" Age="47" AccountId="338430"/>
<row Id="776" Reputation="101" CreationDate="2013-05-21T04:15:01.000" DisplayName="[illegible]" LastAccessDate="2014-01-04T05:01:21.860" WebsiteUrl="http://www.gyswebdesign.com" Location="Abbotsford, Canada" AboutMe="Currently going to the University of the Fraser Valley for my Degree in Computer Information Systems." Views="0" UpVotes="0" DownVotes="0" Age="27" AccountId="8194"/>
<row Id="784" Reputation="133" CreationDate="2013-05-21T05:46:56.000" DisplayName="[illegible]" LastAccessDate="2015-03-20T16:10:07.953" WebsiteUrl="" AboutMe="<p>Linux Systems and Networking; Mostly the the intersection of Networking with Linux systems.</p> " Views="0" UpVotes="0" DownVotes="0" AccountId="2794758"/>
<row Id="794" Reputation="370" CreationDate="2013-05-21T07:57:06.000" DisplayName="[illegible]" LastAccessDate="2013-08-28T08:36:23.587"

FIGURE 4.3: Sample social networking data

```
<row
    Id="89"
    Reputation="1173"
    CreationDate="2013-05-07T21:09:29.000"
    DisplayName="                    "
    LastAccessDate="2016-02-05T17:05:02.720"
    WebsiteUrl="http://xenith.org/"
    Location="Sacramento, CA"
    AboutMe="<p>I'm a Network Engineer currently working for Hurricane
        Electric. I specialize in service provider networks, IPv6, and network
        security.</p>"
    Views="1"
    UpVotes="6"
    DownVotes="0"
    Age="34"
    AccountId="1078694"
/>
```

FIGURE 4.4: Sample extract from social networking data

As shown in Figure 4.3 one row – which is reformatted for better display and shown in Figure 4.4– contains attributes *id*, *reputation*, *location*, *aboutme*, *views*, *upvotes*, and *downvotes* among others. These features contain useful information to boost the relevance of a job seeker profile when combined with resumé. Because they contain more information about the job seeker than the resumé contains. For example, *reputation*, *upvotes*, and *downvotes* are parameters associated with how others valued the contribution, e.g., answers of the user on social network.

Type of Data	**Data Size**	**Use in the Research**
Online resumé	783,000	provides useful information for job seeker modeling
Web survey	1,700	used to study the challenges in job seeker data collection which, in turn, is used as input for the development of DTF
Social networking data	2,300	enrich the data obtained from resumes to better model the job seeker

TABLE 4.1: Data used for job seeker analysis and modeling

Table 4.1 shows type, size of the data used for job seeker analysis and modeling together with its purpose. Online resumé, and web survey data are used for providing information about job seeker from the job seeker's own perspective while social network data provide information about job seeker from the perspective of others in his network.

4.1.3 Preprocessing and Integration

Integrating the data collected from various sources require processing the raw data to produce a collection of job seeker data that contains a sequence of characters. There are a number of issues to address during integration:

i) Format of the job seeker data varies from source to source. For example some job seeker resumés are available in formats such as pdf, doc whereas, others are available in html, or xml formats. Extracting relevant content from these formats is done using various scripts written for this purpose.

ii) Encoding of the documents is another difficulty that needs to be dealt with during data integration in order to store the data in a uniform encoding. This requires conversion of the data from one encoding to another. In this study, the data was stored in UTF-8, i.e., all the data that was in UTF-8 is kept as is and others such as ISO-8859-1 were converted to UTF-8.

iii) During data integration, language in which the job seeker information was stored was also an issue. Job seeker data that was collected in multiple languages. This study limited its scope in working on resumés that are only in English because of its wide usage. Automatic recognition of language was performed to exclude all non-English resumes.

iv) Integration of data from variable sources has different challenges related to preserving privacy (such as inconsistent structure, private information, etc.) that requires thorough consideration. Personally Identifiable Information (PII) refers to "information which can be used to distinguish or trace an individual's identity either alone or when combined with other information that is linkable to a specific individual." (Krishnamurthy and Wills, 2009)

PII is the most central concept in information privacy regulation and privacy laws typically turn on whether privacy harm is involved because the assumption behind the privacy laws is that "if PII is not involved, then there can be no privacy harm." (Schwartz and Solove, 2011) Thus, to ensure that private data is

obfuscated, PII, i.e., name, email and address, in every record of the data is replaced with automatically generated content without losing the semantic of the data contained in the record. This was done by scripts designed and developed specifically for this purpose during the course of this research. The resulting dataset was fully anonymized and obfuscated personally identifying information like name and email while maintaining the quality of the original data. Integration of these diverse data involves extensive text analysis task of refining the text by: i) pruning unnecessary characters used in formatting job seeker resumé; ii) pruning words, i.e., html tags used only for formatting purposes and are irrelevant to the description of the content of job seeker profile; iii) pruning stop words, i.e., too frequent words that do not add value in discriminating the job seeker from the others, if used to represent the data; iv) eliminating too few words as these may be proper nouns that do not add significant value in describing the job seeker and hence not good at discriminating the job seeker from the others; v) dealing with missing data that happens as a result of data integration from different sources. Missing data happens due to structure mapping, i.e., a feature that exists in one of the data may be missing in the other, and thus in the resulting integrated data leading to missing value (NA) that needs special treatment in the analysis; and vi) deduplication that results from collecting multiple instances of online job profiles. This redundancies lead to unnecessarily large data without significant value addition. For that reason, deduplication – a preprocessing task aimed at removing job seekers that have multiple instances in the dataset – is performed in order to reduce the collected data of job seekers into a set of unique job seekers.

4.2 Job Seeker Analysis and Modeling

The initial step in job seeker analysis and modeling involves representation of job seekers profiles which will be used as an input to job matching system. After extracting job seeker profile from candidate profiles, job seeker is described as a list of features each of which are composed of features (cf. Figure 4.2), i.e., list of list, such that job seeker is defined by the following hierarchical features: i) skills that contain list of skills, ii) education containing list of educations and trainings achieved by the job seeker, iii) experience having list of work experiences, iv) preferences containing list of specific preferences of the job seeker.

Representation of job seeker by finding an optimal subset from a set of features requires a number of preprocessing steps to select words that best describe the job seeker skills, education, and experience among others, i.e., feature selection when applied to document representation is selection of statistically

significant words that better discriminate one document from another. Because, on one hand, having few features makes it difficult to formulate a viable hypothesis. On the other hand, having features that do not have the capacity to discriminate documents from one another adds noise, i.e., documents that should not have been matched will be matched.

In order to represent job seeker using the terms that describe it, let T be set of terms in the vocabulary of the dataset consisting of $t_1, t_2, \ldots, t_n$ as selected feature terms that better describe the job seeker with feature weights $f_1, f_2, \ldots, f_n$ and C be the set of candidate job seekers in the dataset containing job seekers $\vec{c_1}, \vec{c_2}, \ldots, \vec{c_m}$ with features $f_1, f_2, \ldots, f_n$ as shown in Equation 4.1.

$$\vec{c_i} = \{f_{i1}, f_{i2}, \ldots, f_{in}\}, \forall i \in \{1, 2, \ldots, m\} \tag{4.1}$$

$$\text{where:} \quad n = |\text{T}| \text{ and } \text{m} = |\text{C}|$$

Job seeker c (referring to candidate job seeker) feature representation is done using matrix C of m x n, where m is number of candidate job seekers, i.e., number of rows of the job seeker matrix and n is the number of terms in the vocabulary, such that $c_1 f_1$ is the feature weight of the term t_1 present in job seeker c_1, $c_2 t_2$ is the feature weight of the term t_2 present in job seeker c_2, and so on as shown in Table 4.2.

	t_1	t_2	$\ldots$	t_n
c_1	f_{11}	f_{12}	$\ldots$	f_{1n}
c_2	f_{21}	f_{22}	$\ldots$	f_{2n}
$\ldots$	$\ldots$	$\ldots$	$\ldots$	$\ldots$
c_m	f_{m1}	f_{m2}	$\ldots$	f_{mn}

TABLE 4.2: Representation of job seeker using term-document matrix

The job seeker model is then extracted for representation from formats similar Table 4.2 to build the matrix C containing feature vectors of job seeker as shown in Equation 4.2.

$$C = \begin{bmatrix} f_{11} & f_{12} & \dots & f_{1n} \\ f_{21} & f_{22} & \dots & f_{2n} \\ \dots & \dots & \dots & \dots \\ f_{m1} & f_{m2} & \dots & f_{mn} \end{bmatrix} \tag{4.2}$$

Representing job seekers using term co-occurrence matrix using the raw text data makes the matrix too big and the matching computation too complex. To address this, simplifying the matrices using Singular Value Decomposition (SVD) helps to approximate the values of the matrices and avoid the effect of noisy data in the matrices thereby reducing the computational complexities. For example, consider the following dataset consisting of three candidate job seekers looking for *Software Developer Intern*, *Business Development Manager IOT* and *Web Developer* positions:

Job seeker ID	**Job title**
c_1	*Software Developer Intern*
c_2	*Business Development Manager IOT*
c_3	*Web Developer*

TABLE 4.3: Example extracts from candidate job seekers

After preprocessing, removing suffixes and lowercasing the data in this case, we get the dataset vocabulary T = {software, develop, intern, business, manage, iot, web}.

	software	*develop*	*intern*	*business*	*manage*	*iot*	*web*
c_1	1	1	1	0	0	0	0
c_2	0	1	0	1	1	1	0
c_3	0	1	0	0	0	0	1

TABLE 4.4: Example representation of job seeker using term-document matrix

That is, the term *software* occurred in candidate c_1 1 time, and it did not occur in c_2 and c_3. Similarly, the term *develop* occurred in all the candidates' data. This dataset is represented as in the matrix in Equation 4.3:

$$C = \begin{bmatrix} 1 & 1 & 1 & 0 & 0 & 0 & 0 \\ 0 & 1 & 0 & 1 & 1 & 1 & 0 \\ 0 & 1 & 0 & 0 & 0 & 0 & 1 \end{bmatrix} \tag{4.3}$$

Modeling job seeker solely based on the data obtained from resumés is not sufficient to get the overall picture about the job seeker features. More information about the job seeker is needed in order to figure out how the job seeker sees himself and how others see the job seeker. These two aspects are modeled using data sourced through self-assessment survey (cf. Sections 4.3) and from social network data analysis (cf. Section 4.4).

4.3 DTF for Self-assessment Survey

Dynamic Text Field is a user interaction method for data collection, whereby user data entries are either adapted until the user fulfills their information need or abandons the suggestions and enter the data afresh. The matching of job description and vacancy will be used as a search space in the DTF to enrich the suggestion pool. DTF is similar to autocomplete systems that are available in browser cache and database, except that it does machine learning based autocompletion. DTF in browser cache is query based, i.e., the previous user entries are cached in the browser for future use. For this reason, a query has to be posed for the autocompletion to start at all and it can be slow for large data. Database trigger-based auto-completion,

on the other hand, data input based on structured (hierarchical data), i.e., data is retrieved from the database with the like of partial input. In database trigger-based auto-completion will not start unless the data has exact match in the database. Machine learning based auto-completion is model-based from input based on unstructured data and works with partial matching to initiate auto-completion.

Job matching is more complex than other matching problems such as people-to-people matching (Pizzato et al., 2010) because of i) inability or resistance of job seekers to provide sufficient data about themselves, ii) difficulty in modeling job seekers and vacancies and iii) complexity of matching process itself. Unlike people-to-people matching where persons provide a lot of information about themselves, including their very detailed personal information, people resist to provide information that vacancies (or matching process) may require. Certainly, if a system could assist job seekers in their task of information provision so that they experience a painless data entry, and then use that information to match job vacancies to seekers, it would have better quality data for a better chance of matching the right job seeker to vacancy.

Studying myriads of web-based data collection systems for purposes ranging from customer relationship management to research shows that capturing data from users using web-based system utilizes open-ended and/or closed-form data capturing devices. That is, these systems use either open response formats or closed formats or a combination of them for capturing data from users. In open-ended format (e.g., text boxes), users can enter anything, whereas in closed-form (e.g., multiple choice lists), users can only choose from given options to enter data. While allowing users to freely give response without limiting options, open-ended components pose high pressure on the respondent-side and have higher processing cost. On the other hand, closed-form data capturing components store data that is less challenging to process. The problem with closed-form elements is that they pose pressure on the designers of user interface. They also limit the options of the user during data entry which in turn may lead to dropout.

While allowing users to freely give response without limiting options, open-ended components pose high pressure on the respondent-side and have higher processing cost (Reja et al., 2003). On the other hand, closed-form data capturing components pose pressure on the designers of user interface and limit the options on the user side while they store data that is less challenging to process.

After a close observation of the effect of web servey data in improving job matching result – DTF – a data collection system that assists user to fill web survey is designed, developed and tested. Use of DTF in web-based data collection mitigates the disadvantages of open response formats and closed formats and maximizes the benefits of both.

Because it keeps the balance between limiting the number of choices and giving freedom to the end user to decide what data to enter. Thus, the user is given options to choose from, while at the same time he is still free to enter what he/she wishes even if recommendations appear. This is what makes it different from a mere autocomplete system.

Development of DTF was achieved using real-time data transfer using AJAX technology and analysis of content collected using NLP as shown in Figure 4.6. Datalist tag of HTML5 and AJAX technology enable retrieval of data from database to user interface towards assisting the user in data entry and leverage the benefits of open- and close-ended user interface elements for data input. Data transfer between the client and server needs to be done asynchronously to ensure a rapid response without the user needing to do any operation to transfer the data, thus improving the user experience. This interactivity can be done using datalist as well as AJAX.

One method of achieving dynamic interactivity of user interface elements with data already in our database is to implement the user interface using the datalist tag of the HTML 5. While this may be relatively easy and straightforward, it is subject to a number of limitations. First, using a predefined tag - with no exception to datalist - gives no flexibility. Second, as shown in Figure 4.5, not all versions of all browsers support datalist (Deveria, 2016). For instance, a closer look at 4.5 reveals that Internet Explorer (IE) version 10 and up support datalist only partially. Furthermore, Figure 4.5 also shows that only a few browsers run on mobile devices support it, i.e., only Android Browser versions 4.4.4 and 37 support datalist. The vast majority of browsers that are widely run on mobile devices (e.g., iOS Safari and OperaMini) do not support this element. This leads to the conclusion that datalist cannot be used if we intend to reach the wider community of users. An alternative to datalist, which has limited support, and

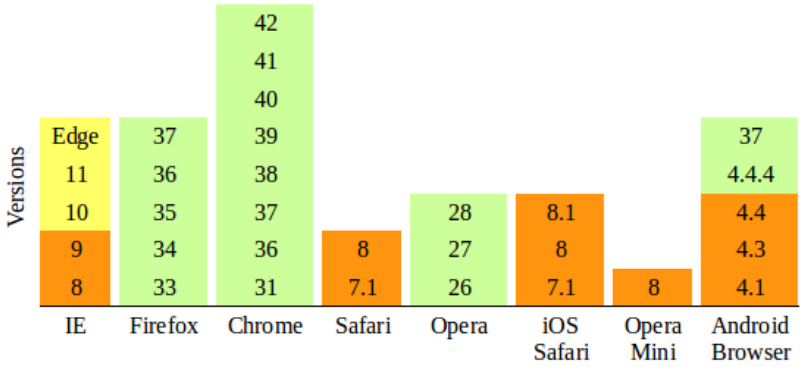


FIGURE 4.5: Browser support of Datalist, [Adapted from (Deveria, 2016)]

thus accessibility (Gibson, 2006), of major browsers, AJAX (Asynchronous Javascript and XML) has wider usability and support (Pilgrim, 2013, Tonkin, 2006). Big Internet companies such as Google®, Amazon® and Yahoo® use AJAX to provide Rich Internet Applications (RIAs) (Fraternali et al., 2010, Gibson, 2006). Moreover, it provides more implementation flexibility on the part of programming the user interface than datalist.

In AJAX technology, the user interface element or its content is modified based on the content of the data in the server behind-the-scene and without an explicit request from the user (Mahemoff, 2006, Paternò et al., 2009). The redrawing of the particular user interface element or its content using what is retrieved from the database on the server guides the user in his/her data entry by suggesting list of choices. The implementation is achieved in our system by taking the partial entry from the user and performing analysis of the existing data vis-a-vis the partial entry and making the suggestions redrawn in the user interface as depicted on Figure 4.6. As shown in Figure 4.6, after the user opens the browser on the client side

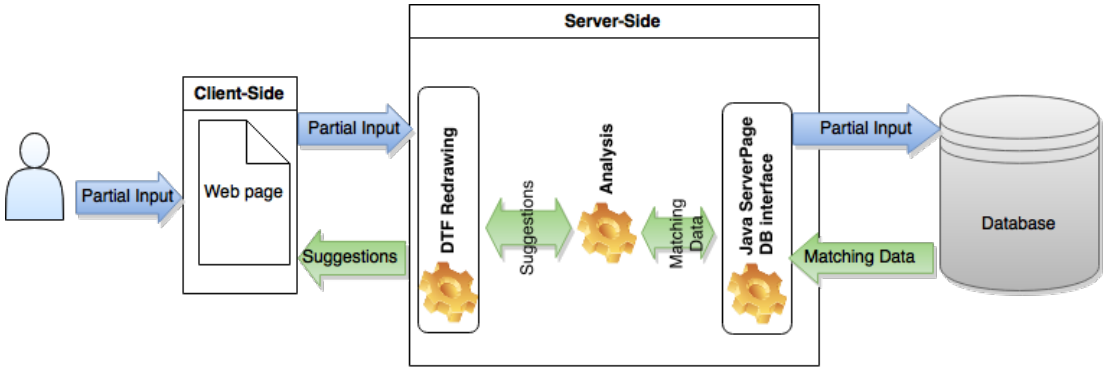


FIGURE 4.6: Implementation of Dynamic Text Field

and gets a web page that has a DTF implemented into it and begins typing text, the partial input is sent to the server side application. Within the server side application, the partial input is sent to the database interface which uses the partial input to fetch matching data. Then the server side application fetches the matching data from the database (which stores the words, distances between words, and co-occurrence statistics of words), performs analysis to determine the ranks and then reorders the suggestions. The ordered suggestions are then sent to DTF redrawing to be listed and presented for the user to choose as shown in Figures 7.2, 7.3a, 7.3b, 7.4a and 7.4b, all in the background. When there is no matching, the DTF redrawing will not have anything to provide, in which case the user is still able to type the input.

With AJAX, the web interface communicates with the server that provides the content asynchronously as the user enters data. The specific kind of data that the server provides to the web interface in response

Algorithm 1: Algorithm for calculating word-word similarity – Adapted from (Jurafsky and Martin, 2009)

Input: Input two strings s and t of length m and n
Output: int EditDistance(char s[1..m], char t[1..n])

```
1  for i ← 0 to m do
2      d[i, 0] ← i;
       /* the distance of a prefix string to an empty second string */
3  end
4  for j ← 0 in n do
5      d[0, j] ← j;
       /* the distance of any second string to an empty first string (i.e., when
          user does not enter anything) */
6  end
7  for j ← 1 to n do
8      for i to m do
9          if s[i] = t[j] then
10             d[i, j] ← d[i − 1, j − 1];
               /* no operation required */
11         end
12         else
13             d[i, j] ← minimum of (
14                 d[i − 1, j] + 1, /* a deletion */
15                 d[i, j − 1] + 1, /* an insertion */
16                 d[i − 1, j − 1] + 1 /* a substitution */
17                 );
18         end
19     end
20 end
```

to AJAX is determined by the suggestion component that performs content analysis (cf. Figure 4.6). What kind of data should the server provide to the web interface in response to AJAX is determined by the suggestion component. The suggestion process is done in a two-step algorithm: word-level and phrase-level.

Using word-word similarity, the distance of the partial entry is calculated with Algorithm 1 where the resulting values of the distance between words in the database and the partial input are sorted. Consequently, a list of suggestions are selected and presented to the user.

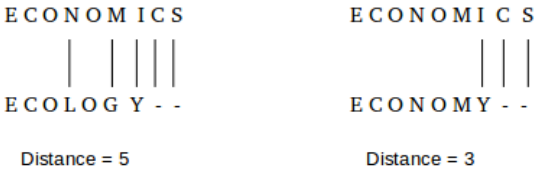


FIGURE 4.7: Word-Word distance: an example

For example, one can see that, with Algorithm 1, the distance between ECONOMICS and ECOLOGY and that of ECONOMICS and ECONOMY, are 5 and 3, respectively, as shown in Figure 4.7. Hence, ECONOMY is closer to ECONOMICS than ECOLOGY. Similarly, if the user enters *devel* as partial input, the distance between the partial input *devel* and all the terms in the dataset is computed. For example, *devel* has a distance of 8, 4, and 5 from the words *software*, *developer*, and *intern* respectively extracted from candidate *c1* shown in Table 4.3. The partial entry is closer to *developer* than *software* and *intern*. As a result, DTF provides *developer* at the top of the suggestion list.

Once the full word suggestion is complete, the next task in the process is the suggestion of subsequent words following the single-word suggestion to make a phrase suggestion. This is done by determining the co-occurrence matrix of every word in the previous entry against the word(s) in the completed partial entry. Klahold et al. (2014) investigated how association mining using co-occurrence of words imitate human word association and found promising result. In order to determine the next word in the suggestion for the user data entry, we need to calculate the co-occurrence probability of the completed user's partial input with the list of available keywords which have been extracted from existing dataset as well as previous entries by using relative co-occurrence frequency as shown in Equation 4.4 (Jurafsky and Martin, 2014). Based on Markov Assumption (Fink, 2014), given the sequence of words $w_1, w_2, ..., w_{n-1}, w_n$, the probability of occurrence of this sequence is computed using Equation 4.4

$$p(w_1, w_2, ..., w_{n-1}, w_n) = p(w_1).p(w_2/w_1).p(w_3|w_1, w_2)...p(w_n|w_1, w_2, ...w_{n-1}) \tag{4.4}$$

Although the probability of the sequence can theoretically be obtainable using Equation 4.4, the probability of longer phrases and sentences is almost always zero. To reduce this, Markov assumption (Jurafsky and Martin, 2014, Lin and Dyer, 2010) is used to approximate Equation 4.4 to get Equation 4.5.

$$f(w_i, w_j) = \frac{count(w_i, w_j)}{count(w_i)} = \frac{count(w_i, w_j)}{\sum_{w_j} count(w_i, w_j)} \tag{4.5}$$

In this notation, $count(w_i, w_j)$ is the count of co-occurrence of the word in the partial input and the candidate next word whereas $\sum_{w_j} count(w_i, w_j)$ is the count of co-occurrence of w_i with anything else in the dataset. For this, we need to define the "window" - which is an indication of how far apart the word entered and the words in the database are. In other words, it is a value that represents how many intermediate words are to be ignored in order to consider two words as "co-occurring". For example in the text from a vacancy data, "skill needed to develop software", "skill" and "software" are considered to co-occur when the window is >=5. They are considered as not co-occurring if the window is <5. Co-occurrence counts are extracted through the analysis of a large corpus of data and the matrix is computed and sorted depending on the values that show how likely they are to co-occur. The resulting matrix, which is structured as shown in Table 4.5 will store a model of co-occurrence of any two words in our dataset. The prediction of the next word generates a list of words sorted based on the maximization of frequencies on each row to which the current word belongs.

	w_1	w_2	w_3	...	w_m
w_1	c_{11}	c_{12}	c_{13}	...	c_{1m}
w_2	c_{21}	c_{22}	c_{23}	...	c_{2m}
w_3	c_{31}	c_{32}	c_{33}	...	c_{3m}
...	...	...	...	...	...
w_n	c_{n1}	c_{n2}	c_{n3}	...	c_{nm}

TABLE 4.5: Word co-occurrence matrix

A vector of fixed length (i.e., a value equal to the number of unique words in the corpus) for each unique word in the corpus is defined. The context vector for each word tells us the number of times other words have co-occurred with the word in the partial input in the given window, e.g. in a window of words, it will check if the other words occurred with the word in the partial input and increment their corresponding element in the context vector. The working of the algorithm proceeds shown in Figure 4.8.

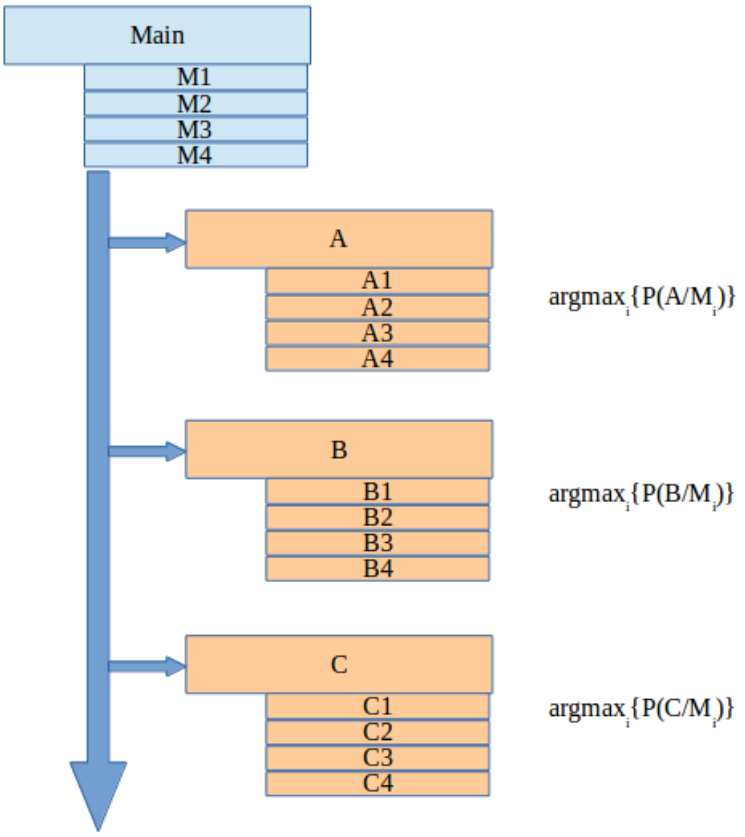


FIGURE 4.8: Context-aware DTF procedures

When the user starts entering data into the first field, DTF assists him in completion of the word and suggestion of the next word. Then the subsequent fields are dependent on the current field and DTF suggests by maximizing the conditional probability of the entry. For example, as shown in Figure 4.8, when the user starts typing in *Main*, DTF first provides lists *M1, M2, M3, M4* from which the user chooses to complete the entry. Then in the subsequent field *A*, *B* or *C*, the top element in the suggestion is the one with the highest conditional probability of occurrence, given the selected element in *M*. This way the DTF keeps on updating the subsequent suggestions according to the context of the data entered.

4.4 Job Seeker Analysis with Social Network

Job seekers who are applying for a particular job vacancy provide their resumé and statement of interest or enter similar information on forms. Matching candidate job seeker to vacancy can be performed using

information provided by the job seeker. In addition to collecting information about job seekers from job seekers themselves, it would be interesting to inquire how matching job vacancies to seekers would perform if a system could gather information about job seekers from other sources like social networks.

Social network is an online communication platform for people who are connected in some ways – be it as relatives, colleagues, classmates or simple acquaintances – expedited by the emergence of IT resulted in facilitation of different communication platforms for people. These days, social networking sites are increasingly becoming key source of determining the behavior of individuals based on their connections, contributions, and reactions. Social networking is playing an important role in the day-to-day life of individuals and organizations. There are a number of social networking platforms being used online with varying focus. Among them, Facebook® (Facebook, 2016), Twitter® (Twitter, 2016), and LinkedIn® (LinkedIn, 2016) are the most widely used platforms for social and professional purposes. Others such as Researchgate® (Researchgate, 2016), Academia.edu® (Academia.edu, 2016) are used predominantly for scientific purposes. Social networks contain more information about job seeker than they state on their resumé. This information is being used during recruitment process. Integrating the data from social networks into job seeker data so that job matching is done automatically can help improve matching accuracy of recruitment systems.

Features that characterize the job seeker are extracted from connections on social networking sites and forums, i.e., from the contributions and endorsements to build template for job seeker modeling. Contributions like questions, answers, comments and "up votes" as well as endorsements are used as means to distinguish the strength of a particular skill from others. For example, using endorsement, as shown in Figure 4.9, this user is more suited to skill *Artificial Intelligence* than *Algorithms*. Because as shown in Figure 4.9 this user possesses the skills *Artificial Intelligence* and *Algorithms* among others. However, it is clearly seen that the user is known by many in his connections as an expert of *Artificial Intelligence* as compared to *Algorithms*.

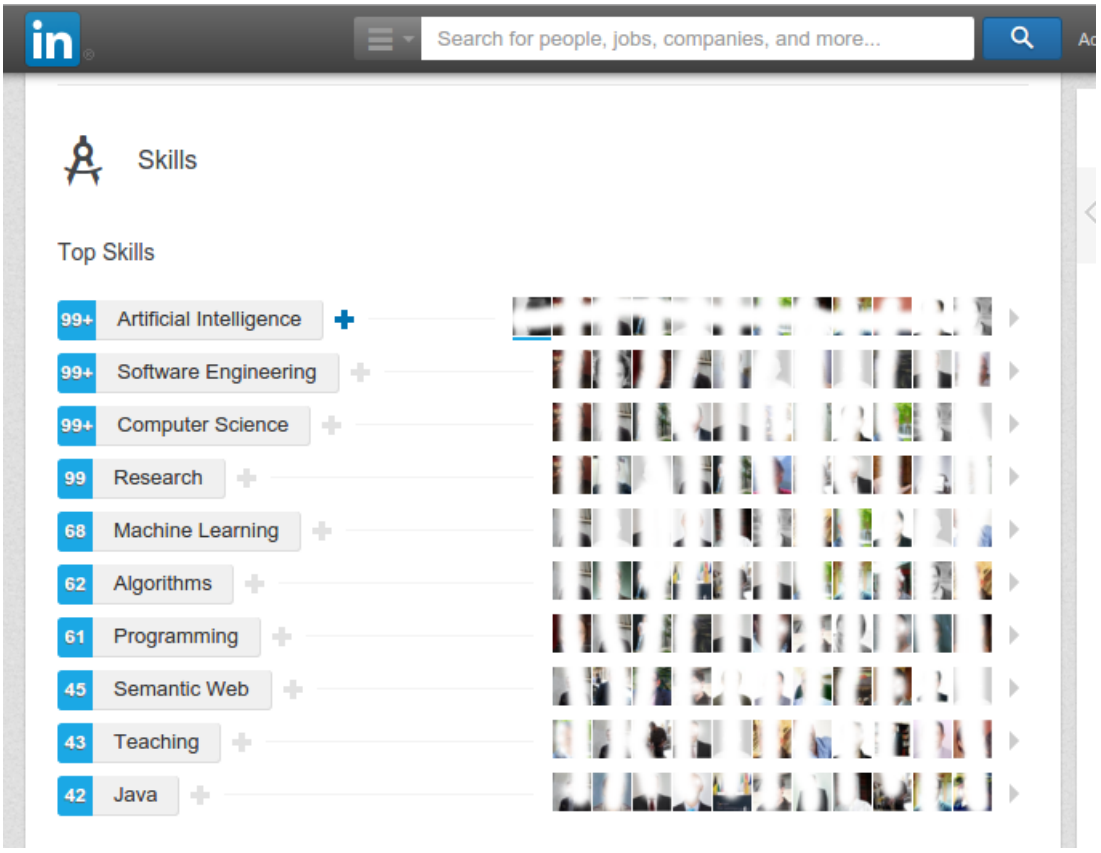


FIGURE 4.9: LinkedIn endorsements of skills[1]

Traditional recruitment involves evaluation of job seeker's training qualifications, personal statement of purposes, and recommendations from supervisors and/or peers. Online recruitment systems seem to lack (Girard et al., 2014) one or more of the key measures (cf. Section 4.5) – self-evaluation and evaluation by others – of a job seeker. However, online systems could incorporate self-evaluation via skill survey and connection evaluation via social networking. That is, considering social networking data enables implementation of the role recommendation letter plays in traditional recruitment to online recruitment systems and helps explore how we can incorporate this information from social networking data into online job matching and recruitment systems.

Recommendation and references (either in the form of letters or otherwise) play a big role in getting more information about (and assessing the background of) a job seeker from the supervisors' or coworkers' point of view. This information can be obtained from social networking sites. These sites are good sources not only of employee profiles, but also their connections together with the witness of their connections about

[1]Part of the diagram is blurred in order to hide personal information of the people who endorsed this user

those prospective employees. Though not integrated in recruitment systems, social networking platforms are also being utilized by employers as main tool to search for their prospective employees. According to Itsyourskills (2016), that conducted a survey on the extent to which social networking sites are used for recruitment, 45% of companies use twitter, 80% use LinkedIn® and 50% use Facebook® to find talent.

Expansion of social networking sites such as LinkedIn® (LinkedIn, 2016) and Researchgate® (Researchgate, 2016) to have millions of users around the world is evident for its importance as source of information about job seekers. For example, in LinkedIn® (LinkedIn, 2016), connections of users provide information about one user in two ways: i) by endorsing skills of a member in the networking sites with which they are connected and ii) by writing recommendation that explains how they evaluate the user they collaborated with. On the other hand, in Researchgate® (Researchgate, 2016) connections of users provide about one user by i) approving questions asked by a particular user, ii) approving answers provided by a user, iii) evaluating the publication of a user, and so on.

Social networks are increasingly being used in recruitment and selection processes and hence can be integrated to recruitment systems per se (Athavaley, 2007). This study includes extracting useful knowledge that emanates from relationships between users of social networking sites – the knowledge of each other's skills, expertise, experiences and attitude – to use in evaluating the match between job seekers and job vacancies.

4.5 Measuring Job Seeker Skill

Evaluating and making good decision on multi-parameter scenario like job searching poses a tremendous challenge (Neffke and Henning, 2013) as it requires collecting as many questions as possible, in as short time, because the nature of the data is volatile, e.g., the searched item can be taken by other competitors. Measuring job seeker skill is important in order to discriminate job seekers matching to job vacancies, i.e., ranking job seekers to vacancies is dependent on the measure of the skills vis-á-vis the requirements in the vacancy (Heckman and Kautz, 2013). Measuring job seeker skill is the main challenge in job matching. The difficulties include age of the skill and technology change among others.

Time is one of the important factors to include in the equation of job seeker-to-vacancy matching. The relevance of a skill varies with i) its age, i.e., the time span between the time of job matching and the

time the skill was acquired and ii) whether the user is actively using the skill on the current job, i.e., the period between the time of job matching and the last time the skill was used or iii) whether the user has used the skill actively for longer period and has not used the skill for short time. In the first case, the older the skill, the less its relevance for the job matching, and hence it will be penalized by a lesser weight as shown in the skill measure (cf. Figure 4.10) because the skill might have faded away with time. In the second case, however, the older the skill the better the relevance will be because the skill has been used for longer period that it is easily available for the job. A job seeker that has the skills acquired a certain amount of time away from the time of job application should have the skill penalized to ensure the skill recency and reduce the prerogative of using a certification for its lifetime. Because the value of a certification degrades with time (Guion, 2011, McGill and Dixon, 2013), there needs to be a scheme for pro rata representation of its rate of degradation with the whole duration. Self-assessment is incorporated with a certain scale range where the job seeker estimates how relevant the skill is for that particular time.

The other reason we need skill measuring is technology change. Technology change makes some skills valuable and others obsolete. Skill mismatch can occur "because new technologies frequently require specific new skills that schools do not teach and that labor markets do not supply" (Bessen, 2014). Bessen (2014) showed the difficulties in measuring worker skills that are related to technology. Measuring skill of a job seeker developed in this research involves assigning weights to skills of individuals based on: subjective measure, semi-subjective measure, and objective measure as explained below.

- Subjective measure – skill of a user can be assigned weights based on self-assessment by job seekers themselves, i.e., by letting the user provide the extent of his/her skill on scale. This measure only takes values assigned to skill by the user's own assessment based on how he/she feels the relevance of the skill. This is subjective because the user does not have any objective ways of determining the skill level.

- Semi-subjective measure – semi-subjective measure is done by using i) rated measure using information on recency and duration of experience/training, ii) rated measure using endorsement of skills, e.g., recommendation by networks of the individuals. The more job related professional certification the user possesses, the higher is the value of semi-subjective measure. Similarly, the value of semi-subjective measure of a skill increases with the number of endorsements of a skill on social network.

- Objective measure – objective measure is when administering a short quiz for each skills to the job seeker to determine the skill level. Obtaining objective measure is assumed to provide a more accurate measure of skill relevance. However, it requires more resources, i.e., expertise and time. Administering the quiz requires setting questions to assess each skill a job seeker possesses.

Skill weighting is computed as shown in Figure 4.10 using the three measures, namely, score 1, denoted by x_1 , measures the subjective evaluation of the skill level provided by job seeker's self-assessment; score 2, denoted by x_2, measures the semi-objective evaluation of the skill level of a job seeker calculated from the assessment of his/her collaborators in social network; and score 3, denoted by x_3, measures the objective evaluation of the skill level of a job seeker calculated from the results of quizzes designed for assessment of skills of the job seeker.

As shown in Figure 4.10, subjective (i.e., x_1), semi-subjective (i.e., x_2) and objective (i.e., x_3) skill measures together with their associated weights are provided as input to the perceptron that in turn calculates the cumulative measures of a particular skill as the sum of the product of the score and the weight associated with it. The result of the aggregation is useful for determination of whether the job seeker's skills are relevant to job vacancy or not and to what extent.

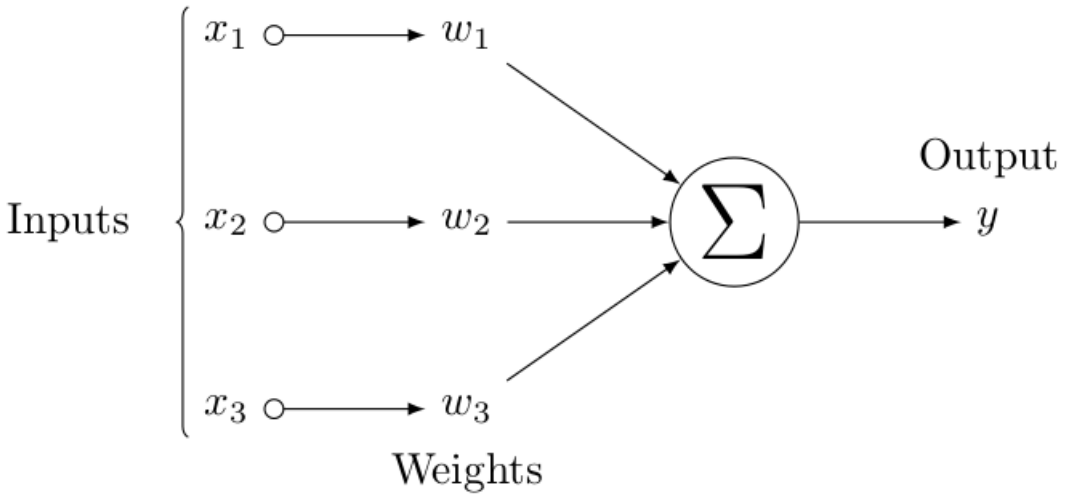


FIGURE 4.10: Aggregation of skill weights

Ranking and weighting provide means of how the features are weighted. By giving weights to each skill and preference of a job seeker, one can collect information about skills and preferences of the job seeker to get a quantitative representation of the job seeker in order to find the rank the job vacancies recommended

in the order of their appropriateness to the skills and preferences of the job seeker. These weights initially come from various sources. First, they can come from the user (i.e., by self-evaluation) such that the user will give estimation of weights to the features that are selected according to the skill/expertise levels. Second, they come from recommendations and social networking sites endorsements of skills of the job seeker. Third, they can be determined by assessing the user on sample tests that measure the level of knowledge on the skill. Forth, they can be determined by data from trainings, and certificates obtained. These sources are then subject to further testing using the recency of the trainings, and certificates because the skill obtained by these trainings or confirmed by these certificates fade away with time (Anderson, 2000, Kahana and Adler, 2002, Kluge, 2014). Thus, the more recent the attended trainings are the higher the weight of the feature. Finally, they can also be determined by the contributions the job seeker made in the form of publications, or answers he provides for questions asked by others in the network on online forums. These contributions get more weight when they have more up-votes or more citations.

The values of endorsements are normalized to be within the range between 0 and 1 in such a way that the skill that does not have any endorsement will be rated 0 and the skill with the maximum endorsement will have skill rate of 1. These ratings are local to the single job seeker as they will not be used to compare a job seeker against other job seekers. Rather they will be used to measure the degree of match between the job seeker and job vacancy for which the job seeker has a relevant skill. These weighted features model the user as a job seeker to determine the most-fit job vacancy to recommend according to the cumulative results from the weights of the features.

Chapter 5

Job Vacancy Analysis and Modeling

In job matching, vacancy analysis not only plays a central role but it also poses tremendous challenges. This is because of information overload (Levitin, 2014), i.e., availability of too much information that creates difficulty on users to make decision as to whether or not a job vacancy is relevant to a job seeker. In addition to understanding and modeling job seekers (cf. Chapter 4) – as one side of the coin in job matching – understanding and modeling vacancy – the other side of the coin – is of paramount importance in order to decide a given job vacancy is relevant to a job seeker. This chapter presents the approaches of understanding and modeling job vacancies through application of data-intensive techniques on occupational standards and online vacancies.

Matching job seeker to vacancy requires learning patterns in the vacancy data that matches with patterns in job seeker data. This chapter focuses on exploring the data and methods necessary to analyze and understand a vacancy, and provides a comprehensive vacancy model as an input for job matching. Prior to matching job seeker with vacancy, the first step is to collect vacancy data, i.e., vacancy advertisements and occupational standard data that can describe job vacancies in good detail. The purpose of collecting more information from occupational standards about job vacancy than simply using vacancy advertisements is to get better matching. While job vacancies tell a lot about the vacancy, occupational standard data adds more pertinent information that are missing in job vacancies.

The collected vacancy data is cleaned in order to remove noisy and inconsistent data. The data from different sources are integrated, i.e., they are combined to form one unified dataset of job vacancy. This dataset is the basis for selecting data that is relevant for analysis. Through utilization of natural language

methods, the vacancy data is transformed into feature matrix of job vacancy containing the features and their relevancy represented by feature weight. Enriching vacancies with occupational standards maximizes inclusion of the vacancy in the recommendation by adding more features. It also helps bolster the weight of existing vacancy features that will then be used to improve the rank of the vacancy in the recommendation.

This chapter presents the process of vacancy data analysis and modeling for job matching. It explains the challenges, scope, data, methods and techniques used in collecting, preprocessing, integrating, analyzing and modeling of vacancies with the goal of modeling it for automatic job matching. It also discusses enriching job vacancies using data from occupational standards.

5.1 Vacancy Data Collection and Integration

In this study the focus is on analyzing vacancies in the area of Internet of Things (IoT). The IoT is a system where a number of electro-mechanical machines, computing devices and other objects as well as human beings that are given a unique identifier, and have the capacity to transfer data over networks without requiring active participation of a human. Vacancy data in IoT is chosen on the grounds of availability of data, dynamic nature of the content of vacancies in the area, and variable terminologies used in the field. More specifically, job vacancy in the domain of IoT are chosen in this research because i) IoT jobs are multi-disciplinary in nature. The skill requirements as well as their description is of highly multi-disciplinary characteristics, i.e., IoT being, a convergence of a number of fields, leads to a requirement of job seekers with multi-disciplinary background (Ma, 2011). ii) IoT jobs reduce data sparsity while simultaneously providing diversity. In the broad view of the job market, it is focused in one particular job – IoT – but IoT vacancies requirements mainly involve IT, engineering, data analytics and administration skills tailored to dealing with the demand of interconnected devices. iii) IoT jobs are emerging jobs and are growing fast. With the trends published by job vacancy aggregators such as Indeed.com (2017), it has been found that IoT job postings as well as job seeker interests towards them is steadily increasing over the last couple of years. Figure 5.1a reveals the growing trends of IoT job posting in the recent years (i.e. in the interval of 2014-2017). This is an indication that IoT jobs are growing and need attention in matching them with job seekers. Similarly, Figure 5.1b depicts the job seekers' interests for IoT jobs which have been drastically increasing over the past years (2014-2017). Comparing the job

postings and job seekers' interests shown in Figure 5.1, one may identify the unmet demand of IoT jobs for job seekers, and thus the need for better matching.

(a) Growing Number of Internet of Things (IoT) Job Posting in Recent Years

(b) Increasing Job Seeker Interest for Internet of Things (IoT) Jobs in Recent Years

FIGURE 5.1: IoT jobs in recent years (Indeed.com, 2017)

5.1.1 Data Source

Data used for vacancy analysis was collected from online job postings. Vacancies were crawled and the data scrapped from the web. The sources of the data are diverse because web crawling was done starting from job aggregators and following the links in order to fetch the vacancy data. The job aggregator used in this research is Indeed (Indeed.com, 2017) based on the popularity it has in Europe measured by its online hit counts (Sundberg, 2017).

In addition to the vacancy advertisements, data for vacancy analysis includes job specifications from standard occupations. There are a number of occupational standards ISCO (ILO, 1997) and ESCO (European Commission, 2015) being the most widely used. The standard job specification of occupations was collected from ESCO (European Commission, 2015). The reason for choosing ESCO(European Commission, 2015) over ISCO (ILO, 1997) is because ESCO is an extension of ISCO and it provides a more specific job titles as shown in Figure 5.2. The ISCO occupation hierarchy has four digits with one layer above the more detailed five-digit ESCO, i.e., occupations are more specific in ESCO as compared to ISCO. Moreover, ESCO further specifies the ISCO occupation groups into with three feature groups: occupations, skills and competences, and qualifications. For example, as shown in Figure 5.2, ISCO level

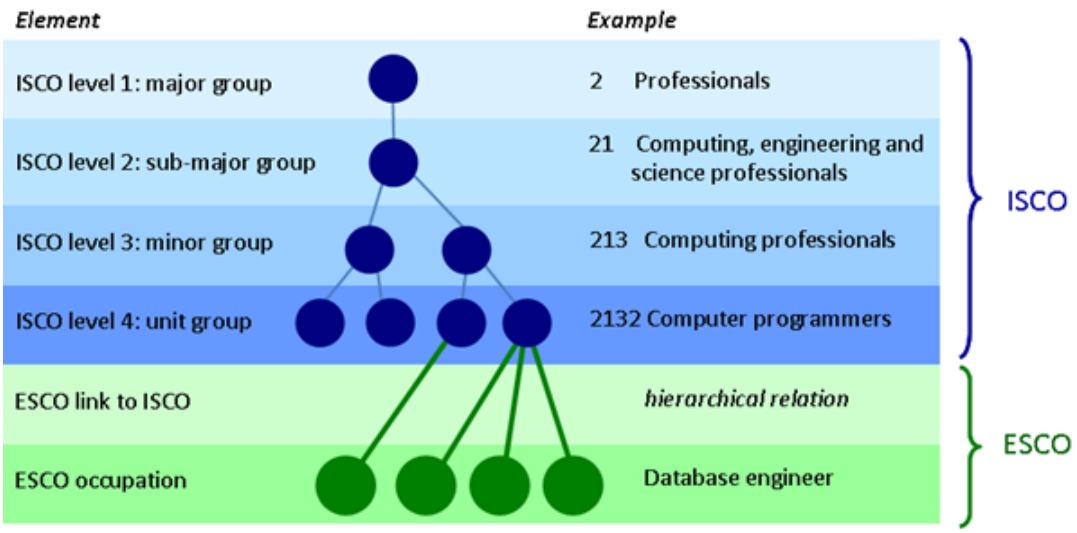


FIGURE 5.2: Relationship between ISCO and ESCO, adapted from European Commission (2015)

1 which is a major group *Professionals* is divided into ISCO level 2 sub major group, one of which is *computing, engineering and science professionals*. Further dividing the sub group in the hierarchy is ISCO level 3 which is a minor group. When *computing, engineering and science professionals* is divided further, it has *computing professionals* among others. The final ISCO level is the unit group, which is the leaf in the hierarchy tree. Thus *computer programmer* is one of the unit groups of ISCO standard. ESCO extends this division into more specialized occupations with specific skills and competences as well as qualifications associated with them, i.e., *computer programmer* occupation is divided further into occupations like *database engineer*. This is specific occupational title that is matched to job titles in vacancy advertisements.

5.1.2 Vacancy Data Collection

Online vacancy advertisements were collected using a Spider developed to scrape vacancy data for a job in the area of Internet of Things using procedure depicted in Figure 5.3. Fetching many pages at once puts a lot of pressure on the remote server and may even take it offline. This can lead to the web scraper identified for overloading the server. In addition, many of the websites do not allow web scraping. Because of these reasons, a socket layer protocol was used to first create handshake connection to the remote server, a random restart of the scraper – with a different identity anonymizing data collecting machine – was used to simulate a human web viewer and avoid being blocked due to frequency of requests. Vacancy data

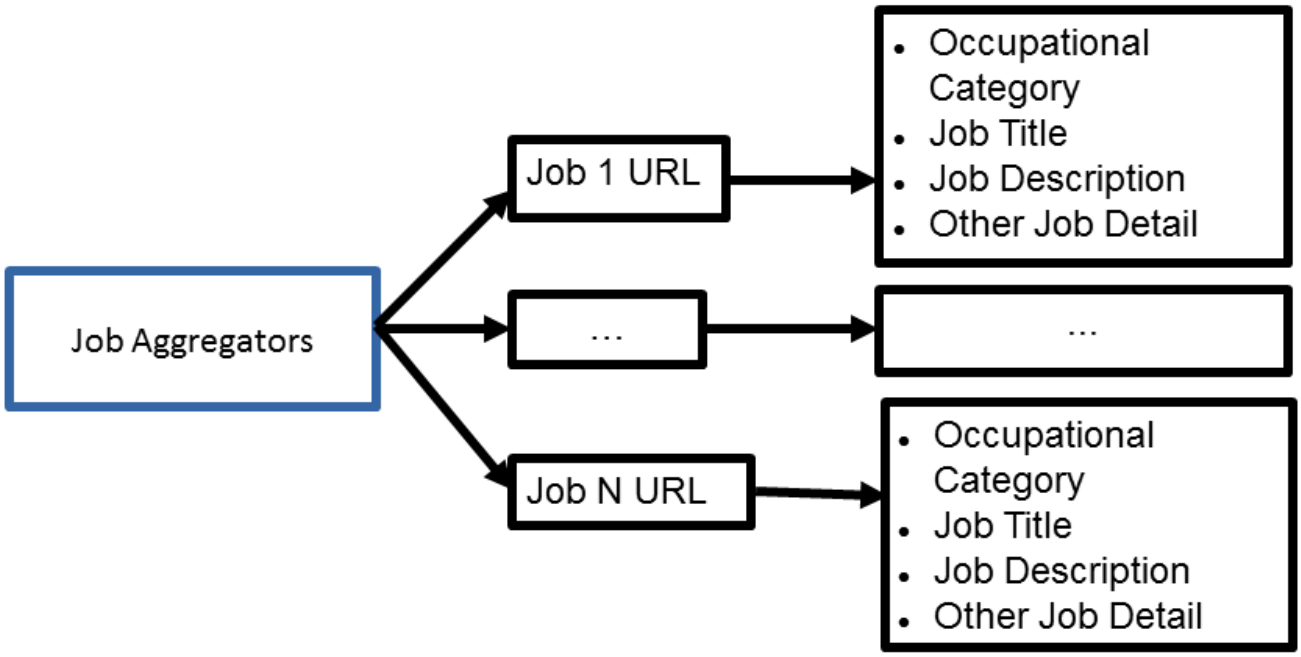


FIGURE 5.3: Vacancy data collection

collection was done in accordance with various ethical web scraping guidelines (Landers et al., 2016) regarding data (i.e., on whether one can take the data, what to use the data for, publish the data) and technological infrastructure (i.e., not overwhelming the remote servers during data collection). First, the terms and conditions of the websites that are used as source of the data have been read and respected.

Second, the following measures are implemented in the crawler to address these issues regarding the identity of the data collector, the developed spider explicitly identified itself in the header with explicit description of the application, source operating system and the email of the researcher.

Third, regarding the pressure on the infrastructure on which the source websites are running, the following measures are taken: i) the spider was tested during office hours with limited number of requests so that the major operation of data collection is performed at off-pick hours of the day and on weekends to ensure that the remote servers are not strained by the crawler leading to denial of service; ii) the spider is designed to pause between requests so that the crawler does not overwhelm the remote server by continuous requests. That is, the spider was written throttled in such a way that it is forced to wait a certain amount of time before it fetches the next web page in the list. This helped limit the rate at which the spider collects the data and avoids unnecessarily straining the remote server.

The occupational standard data that was used to enrich the vacancy data, was obtained from ESCO as it was available for download after provision of researcher details. From the standard occupation data contains detailed information about a particular job title. From ESCO data, one job title has extensive description about the occupation. Skills and competences required for that occupation as well as the qualification requirements for the occupation are stipulated.

Type of Data	**Data Size**	**Use in the Research**
Vacancies	63,000	provides information about job vacancy advertisement
Occupational Information	2,951	enriches data obtained from vacancy advertisements by extending features or boosting weight of existing features

TABLE 5.1: Data used for vacancy analysis and modeling

Table 5.1 shows type, size of the data used for vacancy analysis and modeling together with its purpose. Vacancy advertisement and occupational standard data are used for providing information about job vacancy from the employer's perspective and standard description corresponding to the job title of the vacancy, respectively.

5.1.3 Data Preprocessing and Integration

A) Extracting content – The raw vacancy data collected from online vacancies has a lot of formatting text around the content interesting for the study. One can observe that Figure 5.4, which is partial content of a particular vacancy advertisement, shows data used for formatting and referencing. For example, from Figure 5.4, the interesting data that caries meaning for our analysis are: company represented with variable *cmp*, location denoted by *loc*, *county*, *zip* and *city*, and *title* which stands for job title.

Extraction of useful data from the raw vacancy metadata obtained from job aggregator as well as the actual vacancy data obtained from the company or agency posting the vacancy was performed for each attribute of the vacancy. For example, from the metadata in Figure 5.4, extracting the job title, we get what is shown in Figure 5.5.

```
jk:'f0d611e49a297d52',
efccid: 'caff23281376b83d',
srcid:'c6c40f68ed4ae948',
cmpid:'db9f9b9d28743c59',
num:'0',
srcname:'Amazon.com',
cmp:'Amazon Corporate LLC',
cmpesc:'Amazon Corporate LLC',
cmplnk:'/q-Amazon-Corporate-jobs.html',
loc:'Seattle, WA',
country:'US',
zip:'',
city:'Seattle',
title:'Software Development Engineer \u2013 Internet of Things
       (IoT)',
locid:'1e8a7dce52945215',
rd:'8i0xAbEkuWUhy6dasPEQkfrjdLYrAUDN75hTmQv9FDQ'
```

FIGURE 5.4: Job vacancy metadata

```
'Software Development Engineer \u2013 Internet of Things (IoT)'
```

FIGURE 5.5: Extracted job title

B) Cleaning Data – Data from variable sources has different challenges (such as inconsistent structure, private information, and the like) that requires thorough consideration. The structure of the data is also very inconsistent. For example, the structure of the four job vacancy raw data obtained from open access online sources differ significantly as shown in Figures 5.7, 5.8, 5.9, and 5.10. Integration of these diverse data involves extensive text analysis task of cleaning the text by eliminating: i) unnecessary formatting characters; ii) words used only for formating purposes and are irrelevant to the description of the content; iii) stop words, i.e., too frequent words that do not add value in discriminating the document from others, if used to represent the data, and iv) too few words as these may be proper nouns that does not add significant value in describing the document and hence not good at discriminating the document from others. For example, the above job title has `\u2013` and single quote character (') that need to be removed in order to get the following job title that is composed of natural language words.

```
Software Development Engineer Internet of Things (IoT)
```

FIGURE 5.6: Cleaning extracted job title

C) Missing Data – Missing data occurs due to structure mapping, i.e., because a feature that exists in one of the data is often missing in the other, and thus in the resulting integrated data. This leads to missing value (NA). Dealing with missing data that happens as a result of integration has been taken care of with special treatment in the analysis phase. When the missing value is absent in vacancy advertisement but present in occupational standard, the term is added and used as feature to evaluate the vacancy. However, when the missing value is present in vacancy advertisement and not in the occupational standard for the given job title, it status as missing value is ignored. When a term is missing in both vacancy and occupational standard for a given job title, then a feature value is set to 0.

D) Deduplication – Job vacancies were scrapped from multiple sources on the Internet. It is very likely that a job vacancy appears in two or more of these portals - thus fetched and counted multiple times. This in turn leads to unfairly inflating the importance of a job vacancy gained through unnecessary duplicates of the same data. For this reason, deduplicaiton is performed prior to any data pre-processing task. Determining interval for crawling vacancies affect the quality of vacancy collected and stored, which in turn affects the pre-processing, analysis and modeling. Job vacancies were fetched from the web in two weeks interval. Because crawling in less than two weeks interval resulted in redundancies leading to unnecessarily large data without significant value addition. Some vacancies that stayed for over two weeks were fetched more than once resulting in more than one copy of the same vacancy. Deduplication was one of the processes that needs to be done to handle multiple online vacancies collected, i.e., multiple instances of a single vacancy fetched during crawling. For that reason, deduplication – a preprocessing task aimed at removing vacancies that have multiple instances in the dataset – was performed in order to reduce the collected vacancies into a set of unique vacancies. The other option would be extending the data collection time interval by more than two weeks. In this case, the interval will be so long that some vacancies will be missed in between.

The collected vacancies are compared against one another to determine the level of their similarity. When the similarity of two vacancies is too high, only one of them is taken into account and the rest is discarded. Through this process significant portion of the collected vacancy data was removed and the size reduced.

E) Transforming some features – features such as dates on most vacancies only have the number of days they were online and/or last day for application. In such cases, the date they are published, if available, is obtained through a subtraction of the number of days from today, such that publication-date = today – number of days. If not available, however, it is treated as missing value. F) Vacancy data integration – Difficulty in preprocessing and integration of the vacancy data collected from online sources is manifest in Figures 5.7, 5.8, 5.9, and 5.10. All these vacancies have content for job description. However, the heading for the descriptions is different. For example, in Figure 5.7, 5.8, 5.9, and 5.10, job description is named *job summary*, *description*, *job description* and *description*, respectively. In addition to the names, formatting is also so variable making it challenging to integrate them.

Integration of extracted vacancy data is not only affected by the disparity in naming fields but also with the differences in the number of fields. For example, considering the task of integrating vacancies shown in Figures 5.7 and 5.10 results in data with many missing values. Because extracting content of the vacancy in Seeq Corporation (2016) shown in Figure 5.7 produces a record having fields *job title*, *company*, *job summary*, and *skills and requirements*. Similarly, extracting content of vacancy in Penn State University (2016) shown in Figure 5.10 results in a record having fields *job title*, *company*, *closing date*, *salary*, and *description*. From this one can observe that the two vacancies have only two fields in common for which there will be value. The rest of the fields, i.e., *job summary*, and *skills and requirements* for the former and *closing date*, *salary*, and *description* for the latter will be missing in the integrated dataset. For this reason, integration of the data did not consider these fields, rather the plain text.

G) Clustering job vacancies – Similar job vacancies are clustered into one group in order to reduce computational overhead of working with the whole data. The challenge with this approach is how to deal with new job vacancies. Because data is updated, i.e., new data point is added as we collect online survey about job descriptions and crawl the Internet about new vacancy data, an online machine learning algorithm is needed. Because new data appear and change the weights of already existing parameters, an online learning algorithm is seems to be the most suitable solution for this problem because the knowledge needs to be updated in response to the introduction of new data points. For this reason Stochastic Gradient Ascent (SGA)(Bottou, 2012) algorithm is implemented to address this as it fulfills the requirements: considering the new data point and automatically adjusting the statistics of existing analysis; setting weights to the features, taking into account the new entries; adjusting the weights; and generating a new set of suggestions with a new ranking. Once the clusters of job vacancies are built, the closeness of the newly created vacancies are computed against the centroids of the existing clusters.

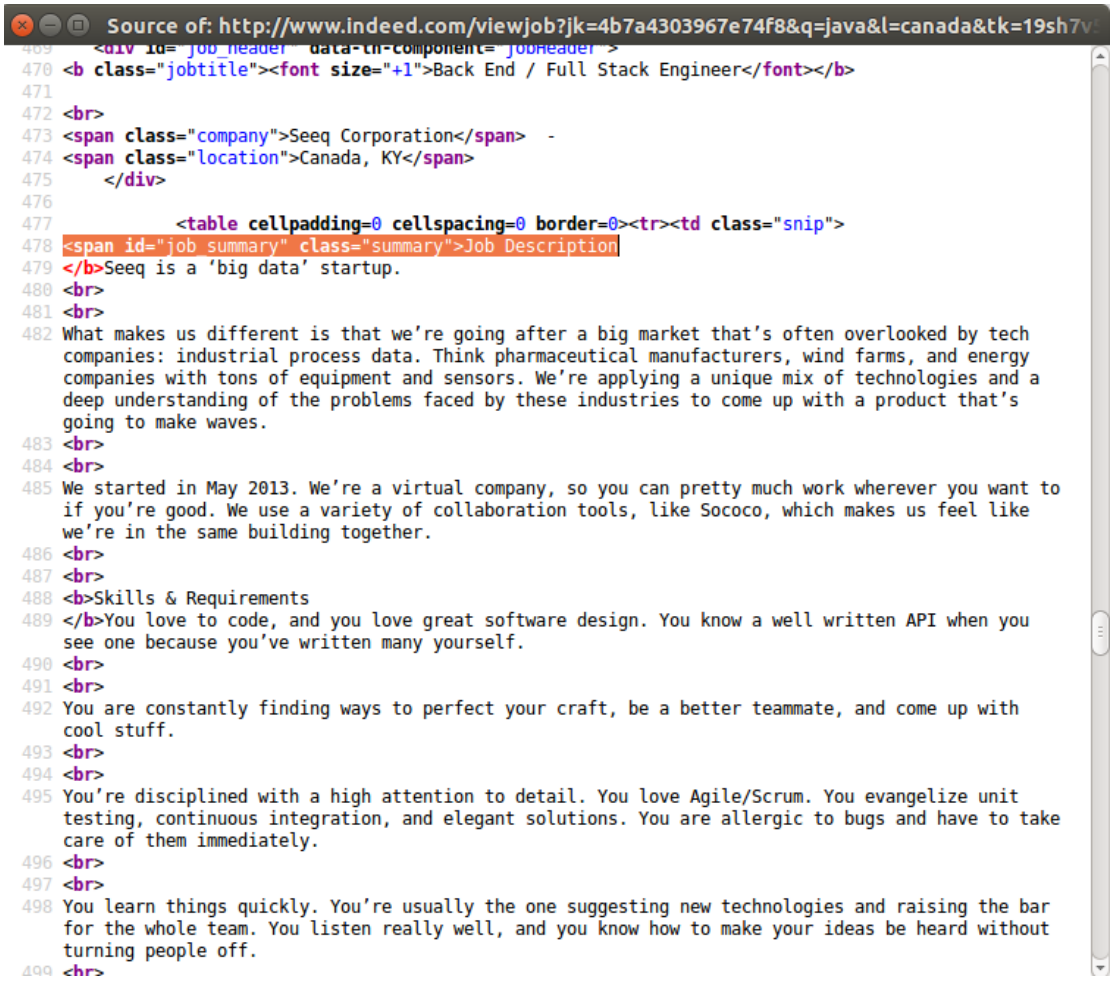


FIGURE 5.7: Structure of Indeed vacancy document (Seeq Corporation, 2016)

Then the new vacancy will be included in the closest cluster, i.e., with which it has the shortest distance from the centroid. The benefit of following this approach is that it helps avoid the costly computation that would arise in recomputing the clusters every time a new job vacancy appears.

On the other hand, the downside of this approach is that newly created vacancy that is too far from every cluster could be placed in a cluster which may not be the optimal cluster only because its distance is the least of all. To address this issue a threshold needs to be determined empirically. This threshold is used to determine if the distances computed between the new job vacancy and the centroids of every clusters is sufficient to assign the new vacancy into one of the existing clusters. That is, if the shortest distance between the centroids of clusters and the new vacancy is greater than the threshold, the new vacancy is considered as outlier and a new cluster is created and it should be assigned new cluster with its own centroid. That is an indication that the vacancy under consideration is an emerging vacancy.

FIGURE 5.8: Structure of ejob-bz vacancy document (Career Evolutions, 2016)

While this approach reduces computation every time a new job vacancy appears and ensures creation of new clusters for handling the newly emerging jobs, it requires the value of the threshold. Yet again, computation of clustering on a regular basis to keep up with the dynamic content is imperative in order to get the clusters as well as their centroids up to date.

Taking example vacancies in Figure 5.11 and 5.12, one can observe that the presentation of the skill requirements vary across vacancy documents. Representing these documents with document by terms matrix, vectorize and transform the documents into the TF-IDF matrix to get a matrix of 2 by 181 dimension. After TF-IDF matrix is built, cosine similarity (Thada and Jaglan, 2013) between documents is computed for the documents, i.e., the number of rows of the matrix (2 in this case) with the tf-idf terms, i.e., the number of columns from the matrix which stands for the size of the vocabulary (181 in this case). This results in array of similarity values [1, 0.34386438]. One can observe that the first value of the resulting array is 1 because it is the measure of cosine similarity of the document with itself.

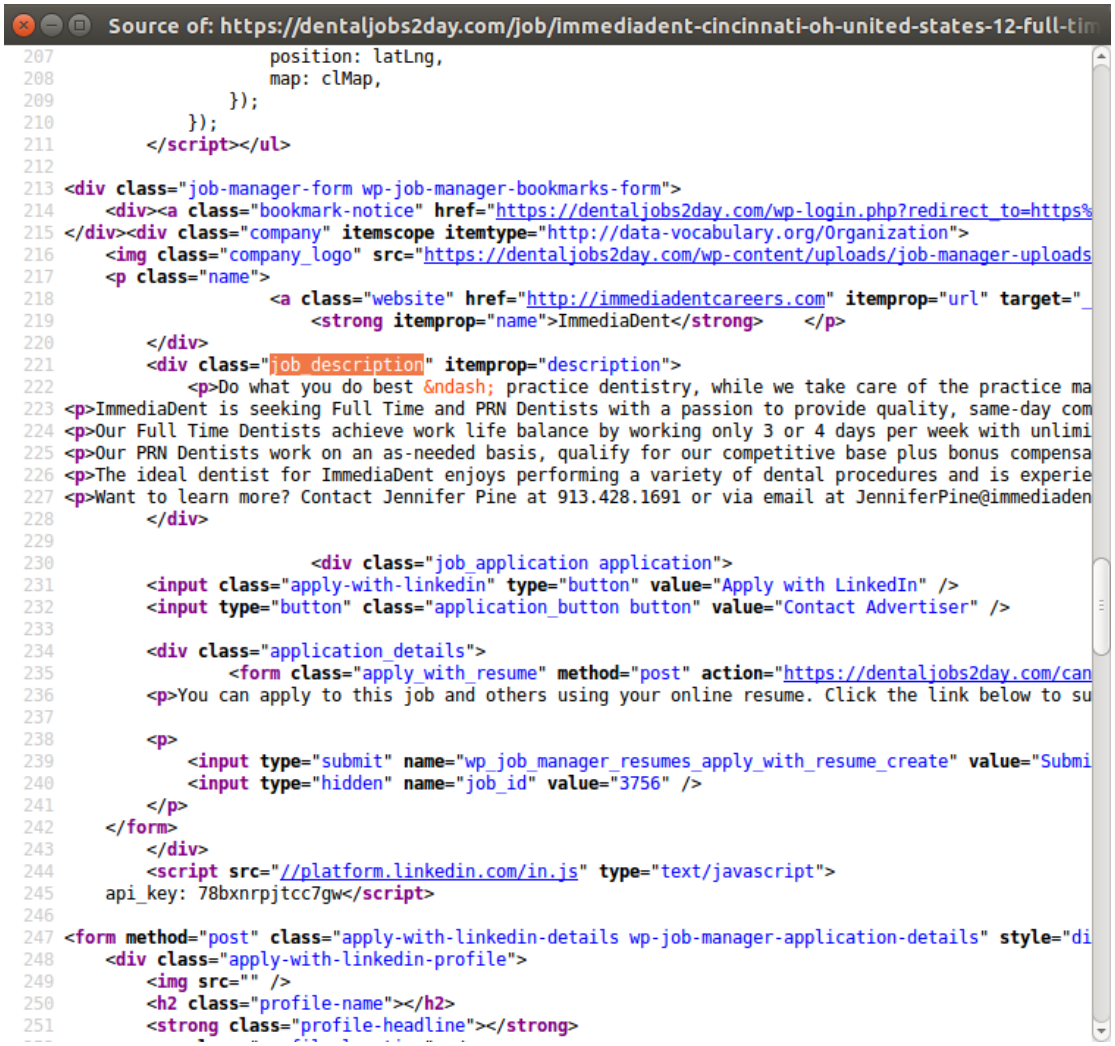


FIGURE 5.9: Structure of jobs2day vacancy document

5.2 Vacancy Analysis and Modeling

Job vacancy is described as a list of features each of which are composed of features, i.e., it is list of lists. Job vacancy is defined by the following hierarchical features: i) job title, ii) job description iii) required skills that contain list of skills, ii) education containing list of educations and trainings required from the job seeker, iii) experience having list of required work experiences, iv) preferences containing list of specific preferences of the employer for the job.

Vacancy features that conform job seeker preferences (such as location) are important features to take into account because studies found that preferences affect job seekers accepting the job. For example, using data from CareerBuilder.com which is the largest job board in the US, Marinescu and Rathelot (2016)

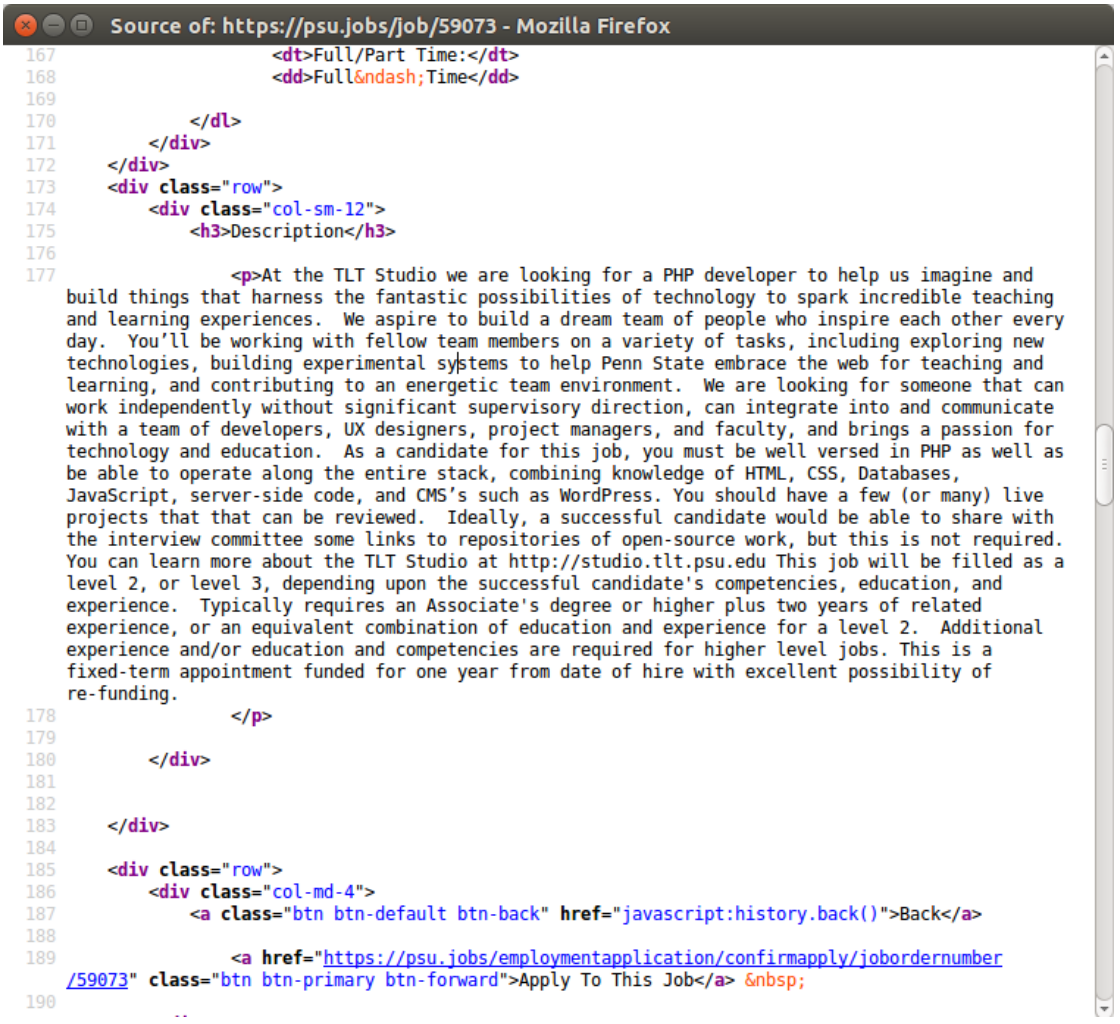


FIGURE 5.10: Structure of psu-jobs vacancy document (Penn State University, 2016)

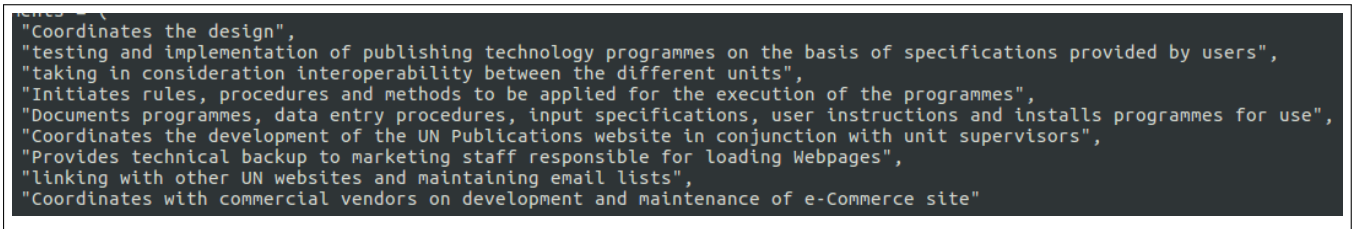


FIGURE 5.11: Example skill requirements for Internet of Things (IoT) vacancy

found out that job seekers are 35% less likely to apply to jobs that are 10 miles away from their residence address.

Prior to feature selection, textual part, i.e., job title and description, of the document representation is done using statistically significant words that better discriminate one document from another. Because

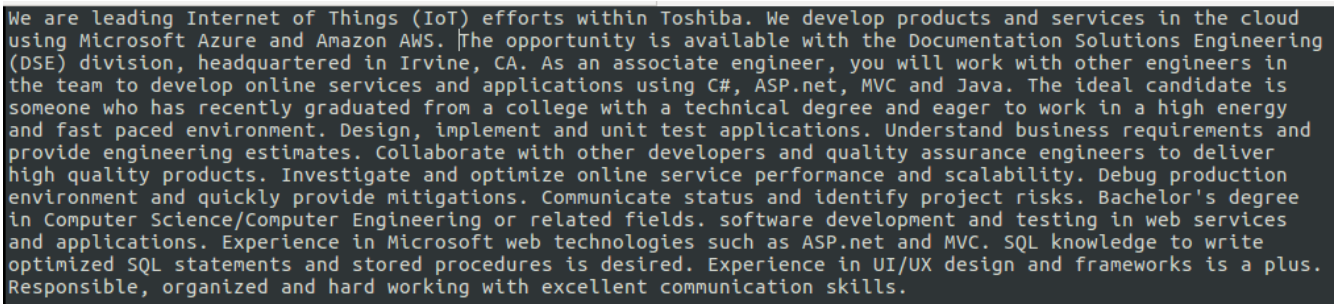


FIGURE 5.12: Example skill requirements for Internet of Things (IoT) vacancy

having few features makes it difficult to formulate a viable hypothesis. However, having features that do not have the capacity to discriminate documents from one another adds noise, i.e., documents that should not have been matched will be matched resulting in reduced precision. Hence, vacancy is represented using statistically significant terms excluding those too few terms that do not have discriminating capacity and those too few terms that create noise.

5.2.1 Enriching Vacancies using Occupational Standards

Vacancies are often not prepared with desired skill sets required by employers (i.e., job provider). Using job title of the vacancy and looking for the closest job description on occupational standards, content of the vacancy can be enriched to have more data from occupational standard of similar job title. Enriching vacancies using occupational standards involves extraction of text from vacancies and occupational standards, splitting the text into words to prepare for the matching process. This method of enriching vacancies with occupational data performs a variety of operations and text analyses related to extraction and matching of several files based on document similarities using indexes. The system is not only helpful to extract and match the specified text from multiple job descriptions but also filters out documents which do not contain a specified text. After the process of matching, the system produces the resulting matched data and stores it into the database for searching.

For instance, considering a job description for a System Administrator, one can find a number of vacancies with different job requirements. Let us take part of the job description $D \in \{D_1, D_2, ...D_n\}$ with content *"... ability to configure systems and networks, manage users, give technical support to users ..."* and two example vacancies V_1 and $V_2 \in \{V_1, V_2, ...V_m\}$ with content *"... experience in troubleshooting systems over wide area network and proven knowledge of working on virtual private networks ..."* and *"... skill in*

corporate email administration and hardware maintenance ...", respectively. It is apparent that some of the key requirements of the vacancies such as *troubleshooting* in V_1 and *maintenance* in V_2 are missing in D. Thus, the system, on the one side, provides suggestion for the job designers to enrich D by incorporating *troubleshooting* from V_1 and *maintenance* from V_2. On the other side, the system provides recommendations to enrich V_1 and V_2 by incorporating the terms *"configure"* and *"network"* from D, respectively (cf. Figure 5.13)

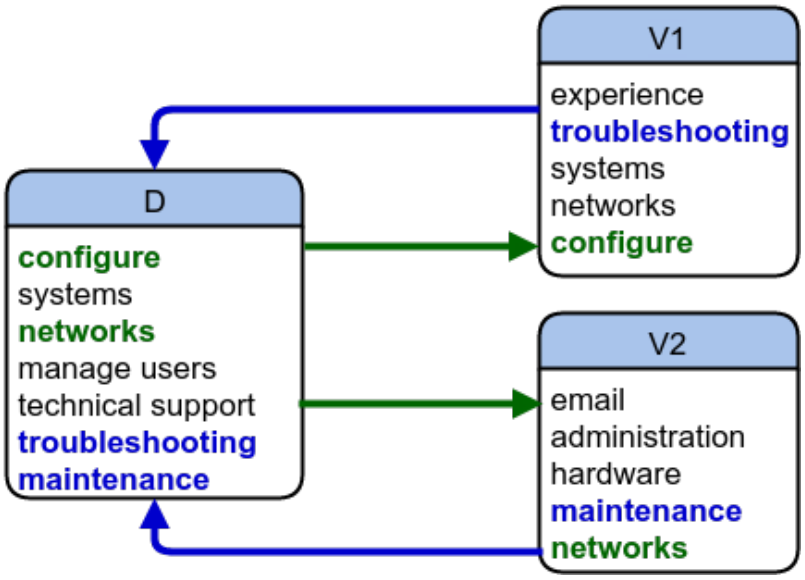


FIGURE 5.13: Enriching job vacancies: an example

5.2.2 Extracting Essential Features from Vacancies

Most vacancies specify the set of skills as required and desired. Required skills (also referred to as essential requirements) are those skills a job seeker must have in order to be considered for the job. Desired skills (also referred to as preferred skills), on the other hand, are those skills which are useful but not essential for the job. A job seeker possessing desired skills would, in theory, excel other competing job seekers who are at par with him/her with the essential skills – giving him/her more chance of being selected. Once the job seeker finds the vacancy interesting, the normal flow of decision process will be that the job seekers will not apply unless they fulfill all of the essential requirements. In its alternate course, i) this deters the would-be-good candidates who after some on-the-job training could acquire the desired skills, ii) job seekers who apply to such vacancies without fulfilling all the required skills could

be selected as long as they possess most of the required and desired skills. Neither of these is favorable matching. This takes us to the need for extracting required and desired skills.

Extracting the required and desired skills in vacancy analysis is performed as shown in Algorithm 2. The algorithm reads the document from the vacancy collection, scans its content and looks for the keywords "Required" or "Must have". These keywords are used in vacancy description to specify which skills/criteria are essential for selecting a job seeker. Likewise, the keywords "Desired" or "Preferred" are used to state the criteria/skills which are possessed by applicant that offer an advantage for selection. However, not all vacancies have the requirements categorized in this form, i.e., required and desired categories are missing in many vacancy data. As a result, from 63,000 vacancies, the algorithm identified that only 990 vacancies contain descriptions with the categories out of which only 123 contain both required and desired skills while 765 vacancies list only required while the remaining 225 list only desired skills. Moreover, not all vacancies are presented in a way Algorithm 2 can filter desired and essential requirements. Some vacancies show desired attributes by stating "is a plus" at the end of each skill, while others specify that possessing the mentioned skill an "advantage." This calls for a modified and more complex rewrite of Algorithm 2.

Though this procedure reduced the number of vacancies from 990 to 123, it improved precision significantly but at huge expense of recall as compared to the baseline. This limits job seekers who would have interest in career switch as any closely related vacancies are eliminated because they did not explicitly state desired/required skills. This adds up on the effect of clustering jobs based on their similarities in limiting career switch. By extracting required/desired skills, some improvements in precision is achieved at the expense of the recall. Reasons are many vacancies are deemed as lacking these categories although they have them in a less explicit way using phrases such as "... is a plus", "... is an advantage", etc. For instance, a job vacancy from Human Capital Research Corporation (2017) in Figure 5.14 shows that desired skills are merged into the required skills with description containing 'a plus'.

Algorithm 2: Algorithm for extracting essential and desired vacancy

Input: Document in vacancy collection
Output: Triplet of Docuemnt, List of Required Skills and List of Desired

```
for Each document in vacancy collection do
    Read through the document;
    if "Required" or "Must have" then
        read text;
        store text in RequiredList;
    else if "Desired" or "Preferred" then
        read text;
        store text in DesiredList;
end
if RequiredList is not empty or DesiredList is not empty then
    return <DocumentId, RequiredList, DesiredList >;
else
    /* Skip the document                                          */
    return ;
end
```

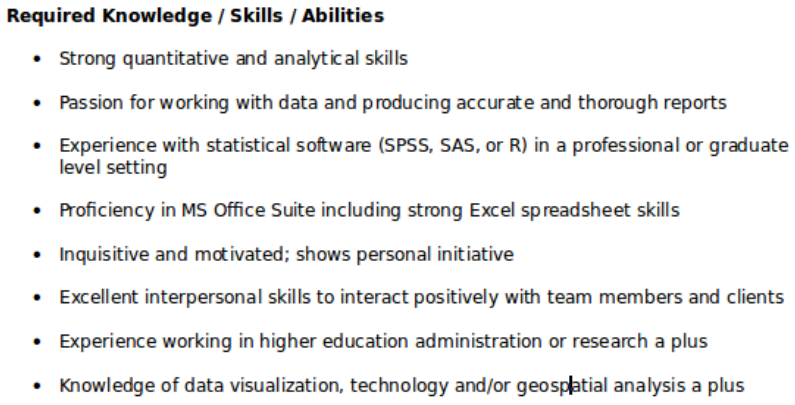


FIGURE 5.14: Challenges to identify required/desired skills (Human Capital Research Corporation, 2017)

5.2.3 Representing Vacancies

In order to represent a vacancy using the terms that describe it, let T be set of terms in the vocabulary of the dataset consisting of $t_1, t_2, \ldots, t_n$ as selected feature terms that better describe the vacancy with feature weights $f_1, f_2, \ldots, f_n$ and V be the set of vacancies in the dataset containing vacancies $\vec{v_1}, \vec{v_2}, \ldots, \vec{v_m}$ with features $f_1, f_2, \ldots, f_n$ such that

$$\vec{v_i} = \{f_{i1}, f_{i2}, \ldots, f_{in}\}, \forall i \in \{1, 2, \ldots, m\}$$

Vacancy V feature representation is done using matrix V of m x n, where m is number of vacancies, i.e., number of rows of the job vacancy matrix and n is the number of terms in the vocabulary, such that $v_1 f_1$ is the feature weight of the term t_1 present in vacancy v_1, $v_2 t_2$ is the feature weight of the term t_2 present in vacancy v_2, and so on as shown in Table 5.2.

	t_1	t_2	...	t_n
v_1	f_{11}	f_{12}	...	f_{1n}
v_2	f_{21}	f_{22}	...	f_{2n}
...	...	...	...	...
v_m	f_{m1}	f_{m2}	...	f_{mn}

TABLE 5.2: Representation of vacancies using term-document matrix

The vacancy model is then extracted from representation similar to Table 5.2 to build the matrix V containing feature vectors of vacancy as shown in Equation 5.1.

$$V = \begin{bmatrix} f_{11} & f_{12} & \ldots & f_{1n} \\ f_{21} & f_{22} & \ldots & f_{2n} \\ \ldots & \ldots & \ldots & \ldots \\ f_{m1} & f_{m2} & \ldots & f_{mn} \end{bmatrix} \tag{5.1}$$

To shade light on how the textual data was represented, consider the following dataset consisting vacancies in the area of IoT:

Vacancy ID	**Job title**
v000	*Software Development and Integration Manager*
v001	*Software Development Manager, IT integration*
v002	*Industrial Internet of Things Manager*
v003	*IoT Software Engineer*
v004	*IoT Saas Architect*

TABLE 5.3: Example extracts from vacancies

After preprocessing, removing suffixes and lowercasing the data in this case, we get the dataset vocabulary T = {'architect', 'development', 'engineer', 'industrial', 'integration', 'internet', 'iot', 'manager', 'saas', 'software', 'things'}. This dataset is represented in the document-term matrix in Equation 5.2 with vacancies as rows, terms as columns and *tf*idf* of terms as feature values.

$$V = \begin{bmatrix} 0.0 & 1.4 & 2.8 & 2.4 & 2.8 \\ 1.4 & 0.0 & 2.8 & 2.4 & 2.8 \\ 2.8 & 2.8 & 0.0 & 2.8 & 2.8 \\ 2.4 & 2.4 & 2.8 & 0.0 & 2.0 \\ 2.8 & 2.8 & 2.8 & 2.0 & 0.0 \end{bmatrix} \tag{5.2}$$

In order to prepare text data for analysis using deep learning, the text needs to be encoded as numbers. Once the documents are split into set of terms and the feature weight of each term is calculated as shown in the document-term matrix 5.2, the textual data is ready to be fed to the deep learning algorithm. Deep learning is done with deep word embedding using the word2vec (Mikolov et al., 2014) algorithm with Skip-gram model (Liu et al., 2015, McCormick, 2016). In the implementation of deep learning for this task, keras (Boschetti and Massaron, 2016) - an open source wrapper for word2vec algorithm is used on the grounds of availability, where the mode is set to *tfidf*. The algorithm allows four modes: i) *binary* which shows the presence or absence of a term, i.e., it is set to 1 when the term is present in the document

and 0 otherwise; ii) *count* is the frequency of occurrence of the term in the document; iii) *tfidf* score of the term in the document; iv) *freq* is the frequency of the term normalized with the number of words in the document.

Chapter 6

Matching Job Vacancies to Job Seekers

Job matching, as defined by Greenberg (2010) is, "*the process of matching the right person to the right job based upon the individual's inherent motivational strengths. It requires thoroughly understanding the job and the person under consideration.*" In job matching, relevance is the goal, while at the same time, it is the unit of measure. Through performing semantic analysis and modeling of both the job seeker and the job vacancy, obtaining and delivering more relevant recommendations is attained, that in turn, improves experience of job seeker in job searching on one hand, and of employers in recruitment, on the other hand.

When matching vacancies to job seeker, the input is a sequence of values for each attribute of job seeker which is obtained from job seeker information collected via web survey and social networking data and a sequence of values for each attribute of vacancy which is obtained from the match between occupation standards and job vacancies. Then the features are aligned. When job seeker feature does not have value, it is filled from social networking data. When a feature is not present in social networking data, the job seeker is dropped to improve data quality. Similarly, when vacancy feature does not have value, it is filled from occupational standard. When feature is not found, the vacancy is dropped to improve data quality. On the other hand, when a feature does not have value for both job seeker and vacancy, the feature is kept in the computation of alignment as it means the two matched by not having value for the given feature. Once the features are chosen, the probability of alignment is compute and the result sort in decreasing order for suggestion. The output of this process is the aligned values for each feature which will be computed to show ranked matches and a cut-off point to decide the n-best matches for the given input.

In an article published on Informer Russell-Rose and Chamberlain (2016), conducted a survey of 64 recruitment professionals and examined the search tasks, behaviors and preferences as well as their most valued functionalities. discussed that information retrieval has not given much attention to using the methods and techniques that proved to be effective in searching on problems of recruitment professionals stated in Russell-Rose and Chamberlain (2016) as "A recruitment professional may spend approximately 27% of their time actively searching for candidates [...], and needs to rapidly evaluate candidate CVs [...]. The activities of recruitment professionals range from directly searching for candidates using job boards through to investigating profiles on social networks to make connections with candidates, as well as gaining broader market intelligence on behalf of clients."

This research is aimed at analysing and developing a jobseeker-to-vacancy matching and recommendation system that is worthwhile for job seekers, employers and/or recruiters, also known as job agents by automating the process of searching job seekers that match requirements of a certain vacancy, analysing the content of their profiles, modeling, and ranking according to their relevance.

This chapter presents the process of bidirectional matching of job seeker to vacancy. It discusses the measure of similarity between the feature vectors of a given job seeker or vacancy with the entries of the vacancy or job seeker matrix, maximizing the similarities, and ranking the result.

6.1 Bidirectional Matching of Job Seeker to Vacancy

Moving from modeling job seeker and vacancy individually to matching them bidirectionally based on similarity of their respective features, i.e., job seeker is matched to relevant vacancies and vice versa, involves feature selection, matching and ranking. In the matching process, one feature of job seeker may be matched with multiple corresponding features of vacancies depending on the learned parameters for features. Feature weighting using deep learning of parameters learns rules to match qualification of job seeker with job title and/or required skills of vacancy; required skills of vacancy with qualification, skills and/or experience of job seeker; preference of job seeker against location and/or benefits from vacancy and so on. It is also the case that features have one-to-one correspondence as in salary (i.e., expected salary) of job seeker matched with salary from vacancy.

Matching is an important step in recruitment process with confirmed economic significance. In 2013 alone, about 25% of firms across the EU had recruitment difficulties associated with lack of staff with

the right skills (Cedefop, 2014). From these 34% reported that they had difficulties because staffs lacked technical competence while 17% said that the applicants lacked workplace competence (also known as soft skills) (Cedefop, 2014). Skill shortages are not the sole reasons for skill mismatch. Unemployment is rising while difficulty filling job vacancies is also rising simultaneously. This leads to the question of whether employees work in jobs that are matched to their qualifications and skills. These mismatches occur due to various factors including: i) job seekers preferences and/or personal circumstances. ii) skill deficit iii) mismatch during recruitment process iv) difficulty keeping up with the rapidly changing skill demand, for example, Cedefop (2015) found out that nearly half of (47%) European workforce, had to adapt to changing skill demand in their job, with about half of them (21%) think that their current skills will be outdated in the next five years.

Cedefop (2008) emphasized that "further research and analysis on the early identification of skill needs" should be within the priorities for research in order to solve the problems of employment mismatches, i.e., shortage as well as surplus.

This model focuses on addressing mismatch that occurs due to lack of matching during recruitment process, i.e., through maximizing the likelihood of matching job seekers who have the requisite skills to vacancy requirements while reducing the chance of those who lack the skills from landing on the job they are not fitting to as described in Algorithm 3 and 4.

In building this model, clustering of similar job seekers and vacancies was done in order to reduce the large data into chunks of manageable size. This improves the efficiency of the matching algorithm (cf. Algorithm 3 and 4) not only by reducing the preprocessing time needed to clean the input data but also by improving matching precision.

Once job seeker and vacancy documents are converted to vectors using terms representation, the document vectors are then normalized with Euclidian normalization (vector length normalization) as in Equation 6.2. Similarity between vectors is obtained by the *cosine* measure between normalized vectors as in Equation 6.1. This similarity measure can be applied on a $M \times N$ *term-document* matrix, where M is the size of the document collection and N that of the vocabulary as in Equation 6.3.

$$sim(d_1, d_2) = v(\vec{d}_1).v(\vec{d}_2) \tag{6.1}$$

$$v(\vec{d}) = \frac{V(\vec{d})}{\|V(\vec{d})\|} \quad \text{where } \|V(\vec{d})\| = \sqrt{\sum_{i=1}^{n} x_i^2} \tag{6.2}$$

Algorithm 3: Algorithm for Matching Job Seeker to Vacancy

Input: Job seeker vector, Vacancy model
Output: List of matching vacancies to candidate job seeker

```
Load candidate job seeker information ;
Preprocess the candidate job seeker information ;
Vectorize the features of candidate job seeker;
Load vacancy model;
for Each entry in vacancy model do
    Compute similarity of candidate job seeker to vacancies;
    /* argmaxv_i Sim(C, V_i), where C is the candidate job seeker and V_i is a record
       in vacancy model                                                              */
    if matching similarity is beyond the threshold then
        store the vacancy record in MatchedJobsList;
        Continue;
    if MatchedJobsList is not empty then
        return <Job Seeker, Vacancy, Similarity >;
    else
        /* Skip the document                                                         */
        return ;
    end
end
```

$$m[t, d] \;=\; v(\vec{d})/t \tag{6.3}$$

In this context, the similarity score of job seeker c and vacancy v is calculated using Equation 6.4.

$$score(c, v) \;=\; v(\vec{c}).v(\vec{v}) \tag{6.4}$$

In bidirectional matching, when we try to match vacancies to job seeker, we use job seeker information as a query to fetch, filter and rank the matching vacancies from the entire collection of job vacancies (cf. Algorithm 3). Likewise, when we match job seeker to vacancies, we use vacancies as queries to fetch, filter and rank the matching job seekers based on the match values (cf. Algorithm 4). The result is ranked according to the match value that dictates degree of relevance of the vacancy to the job seeker or vice versa depending on desired matching direction.

Algorithm 4: Algorithm for Matching Vacancy to Job Seeker

Input: Vacancy vector, Job seeker model
Output: List of matching job seekers to vacancy

```
Load vacancy information ;
Preprocess the vacancy information ;
Vectorize the features of vacancy;
Load job seeker model;
for Each entry in job seeker model do
    Compute similarity of vacancy to job seekers;
    /* argmax_{v_i} Sim(V, C_i), where V is a vacancy and C_i is a record in job seeker
       model                                                                          */
    if matching similarity is beyond the threshold then
        store the job seeker record in MatchedJobsList;
        Continue;
    if MatchedJobsList is not empty then
        return <Vacancy, Job Seeker, Similarity >;
    else
        /* Skip the document                                                          */
        return ;
    end
end
```

6.2 Estimating Similarity between Job Seeker and Vacancies

The similarity between a particular entry in the vacancies matrix and job seekers matrix is a measure of distance between vectors. That is, in order to measure the similarity of a job seeker or vacancy under consideration against the respective matrices, one needs to find the closeness or distance of the vector representation of the job seeker or vacancy from each entry of vacancy matrix or job seeker matrix, respectively – the closest being the most similar.

The matrices contain term weights (TF*IDF) as shown in Equation 3.3 for each document that summarizes the textual data obtained from descriptions of job vacancies, requirements, and other natural language content.

Similarity between the documents was implemented using cosine similarity because it is a measure of distance that takes the length of the document into account (cf. Section 3.6). Cosine similarity, according to Owen and Owen (2012), is a measure of similarity

> *[. . .] that depends on envisioning user preferences as points in space. Hold in mind the image of user preferences as points in an n-dimensional space. Now imagine two lines from the origin, or point (0, 0, . . . , 0), to each of these two points. When two users are similar, they will have similar ratings, and so will be relatively close in space – at least, they will be in roughly the same direction from the origin. The angle formed between these two lines will be relatively small. In contrast, when the two users are dissimilar, their points will be distant, and likely in different directions from the origin, forming a wide angle. This angle can be used as the basis for a similarity metric in the same way that the Euclidean distance was used to form a similarity metric. In this case, the cosine of the angle leads to a similarity value. [. . .] all you need to remember to understand this is that the cosine value is always between –1 and 1: the cosine of a small angle is near 1, and the cosine of a large angle near 180 degrees is close to –1. This is good, because small angles should map to high similarity, near 1, and large angles should map to near –1.*

The cosine similarity measures the cosine of the angle between the vector in question and the vectors in the target matrix as depicted in Figure 6.1. As shown in Figure 6.1, for example, the similarity between candidate job seeker C_1, C_2, C_3 and the vacancy V is the cosine of the angles β_1, β_2, and β_3, respectively. Although the cosine values of angles range between -1 and 1, the value returned from the similarity equation ranges between 0 and 1 since the term weights cannot be negative, thus, the angle between term frequency vectors cannot exceed 90°.

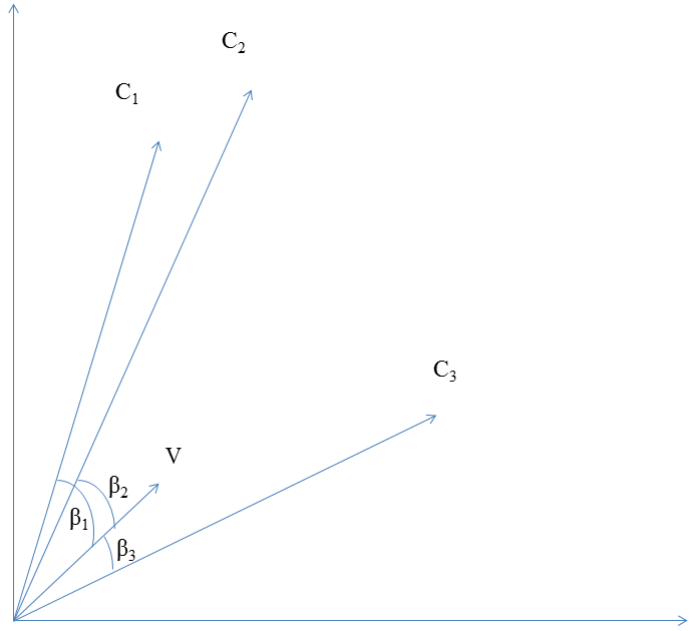


FIGURE 6.1: Similarity between candidate job seeker and vacancy

From a dataset of documents represented by terms t in the vocabulary, a document d can be represented by the weight of each term as:

$$\vec{d} = (w(t_1, d), w(t_2, d), \ldots, w(t_n, d)) \tag{6.5}$$

where: $w(t_i, d)$ = weight of term t_i in d obtained from Equation 3.3

The cosine similarity between any two documents represented by vectors as in Equation 6.5 is calculated using Equation 6.6

$$\begin{aligned} sim(d_1, d_2) &= \vec{d_1}.\vec{d_2} \\ &= \frac{\vec{d_1}.\vec{d_2}}{\|\vec{d_1}\|.\|\vec{d_2}\|} \\ &= \frac{\vec{d_1}.\vec{d_2}}{\sqrt{\sum_{i=1}^{n} d_{1i}^2}.\sqrt{\sum_{i=1}^{n} d_{2i}^2}} \quad \text{where } d_{ji} \in \vec{d_j} \end{aligned} \tag{6.6}$$

Selecting matching text as used in Yu et al. (2014) is applied to find similarity between job seeker and vacancy. The matching returns triplet (C_i, V_j, S_{ij}), where C_i is the candidate job seeker, V_j is a vacancy and S_{ij} is the degree of similarity between the job seeker C_i and vacancy V_j. The bigger the number the more similar the two items are.

That is, matching job seeker to vacancies is maximizing similarities of job seeker to vacancy as shown in Equation 6.7.

$$\arg\max_{sim}(sim(C, \vec{V})) \tag{6.7}$$

Likewise, matching vacancy to job seekers is through maximizing similarities of vacancy to job seekers as shown in Equation 6.8.

$$\arg\max_{sim}(sim(V, \vec{C})) \tag{6.8}$$

Evaluating and making good decision on multi-parameter scenario like job searching requires collecting as many items as possible, in as short time as possible, because the nature of the data is volatile, i.e., the searched item can be taken by other competitors.

The basic assumption is that candidate job seekers have as many common features (with comparable feature vectors) as possible with the vacancy and vice versa. Features are not prepared manually, i.e., features are learned from the data in order to model job seekers and vacancies as vectors, and evaluate the relatedness of each candidate-vacancy pair in a shared vector space. After similarity calculation is performed, the values are sorted and the top N similar results (i.e., the top N entries after the list is sorted in descending order) will be selected as the top N recommended vacancies or job seekers depending on the input vector. In order to rank the bidirectional matches the Vector Space Model (VSM) is used. Ranking based on the watertight vector space model is susceptible to the inherent problem of calculating angle measures (cf. Figure 6.1) as many as the number of vacancies for each job seeker and vice versa. This has an immense computational overhead and it increases with increasing size of the data.

Because sorting the final result is computationally expensive, the matching result is captured on the go, i.e., when the matching item has a similarity score better than any of the sorted TopN items, it will replace one item whose score is just less than the current item.

6.3 Recommendation Processes

Once the matching is performed, recommendation of jobs to job seeker are done in two ways: i) pull recommendation and ii) push recommendation. Pull recommendation occurs when user actively seeks job vacancies. In this case, the system uses the model of job seeker to recommend the list of most relevant ones. Then the system follows the actions of the user, e.g., which vacancy the user selects first, what parameters (about himself/herself) the user modified and so on. This knowledge of the user preferences will then be fed into the model to enhance it.

Push recommendation, on the other hand, occurs when new job vacancies are advertised. The job vacancies will be matched to the list of most relevant job seekers and the list is sent (or pushed) to the user profile (or email). Then the user preference (i.e., the ordered stack of vacancy selection by the user) is captured to remodel the matching.

Chapter 7

Experimental Result and Evaluation

Matching systems have been widely used to help users deal with information overload and make decisions in abundance of choice. Traditionally these systems have been used to recommend items to users. Bidirectional job seekers-to-vacancy recommendation is different from item-to-people recommendation because the interaction between the matching components is done via an intermediary component – job title – which summarizes the expectation of job vacancies and job seekers profiles, on one hand, and the usage of semi-structured data on the other hand. As shown in Figure 7.1, attributes like job title and job description are presented in natural language text – hence unstructured – and require further processing to result in expanded representation.

The job seeker to vacancy matching system, like any system, needs to be tested whether it has achieved the initial conceived expectations. This chapter presents the experimental setup such as the research design, the data used, and methods of analysis. It also presents and evaluates the result of the experiment in job seeker to vacancy matching. Experiments were performed to evaluate the performance of the bidirectional matching system on real data. The result shows the matching job seekers to vacancies with multi-faceted data about job seeker as well as vacancy and compares the effect of each of them on the result of matching.

One task is enriching vacancies with occupational standard data, the second task is a study of data from web survey which led to the development of dynamic text field based user interface design, and the third task is integration of social networking data in job seeker modeling. For each of the tasks the result is presented with comparison of the result vis-á-vis the effect of these tasks.

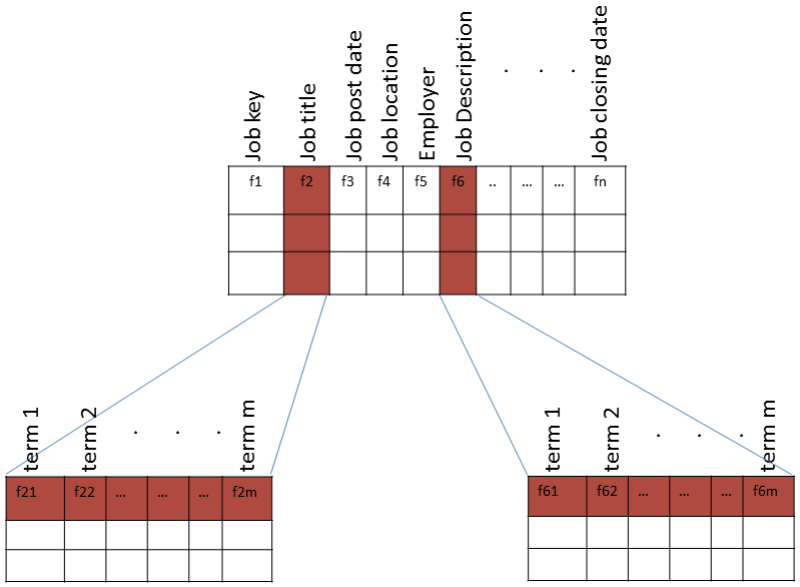


FIGURE 7.1: Semi-structured job seeker and vacancy data

This chapter presents the experimental result and evaluation together with the contribution of the research. It highlights the achievement in the development of the context-aware DTF which improves the user experience in web data collection thereby improving the quality of data obtained from job seeker web survey. It also illustrates the result of job seeker to vacancy matching as well as the improvement of the match when social network data is included to enrich job seeker data.

7.1 Results and Discussion

7.1.1 DTF-enabled Context-aware User Interface

In order to improve the quality of the web survey that helps better understand the candidate job seeker, a context-aware DTF was developed (Chala et al., 2016b). Considering the system conception presented in Figure 4.6, a prototype DTF was trained and tested using data obtained from the Internet mainly ESCO (European Commission, 2015). The result shows the prototype web-based tool using DTF that helps facilitate interaction with user (i.e., job seeker) to collect profiles, assists in the data collection through text analysis and matching techniques and then presents the recommended matching results in ranked order on degree of relevance. The sample result is shown in Figures 7.2, 7.3, and 7.4.

With the implementation of DTF, the user is given a list of suggestions based on his/her partial entries. These entries are used to further refine the suggestions in the subsequent DTFs. For example, if a user in the field of economics starts filling data, with implementation of DTF, he/she is given list of suggestions based on his/her partial input as shown in Figure 7.2. In the sample screenshot shown in Figure 7.2, once the user starts typing partial entry *Ec* in the *Qualification* field, the autocompletion system suggests the list of the closest entries from the model, namely *Economics*, *Environmental protection* and *Electrical Engineering*. Because, these are the entries closest to the partial entry *ec*, the system suggests them with the assumption that the user will select one of them. Once the user selects one of these entries (or provides an entry which is not in the suggestion list), subsequent fields provide suggestions based on the entry provided in this field.

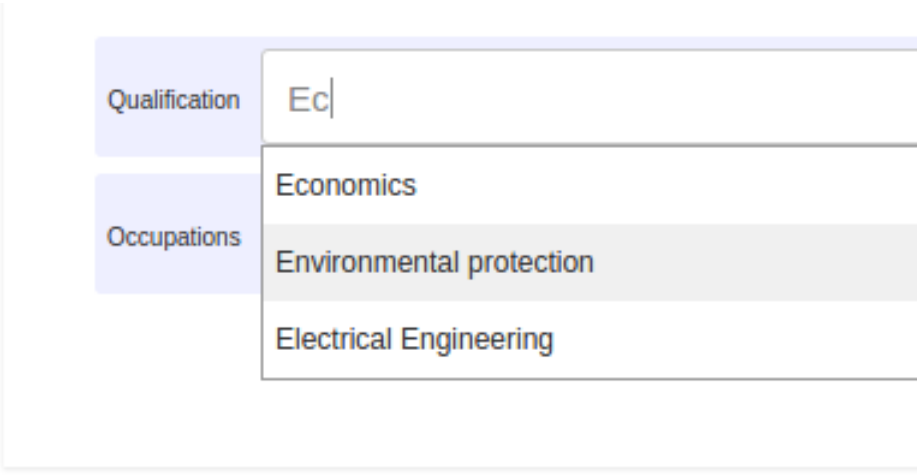
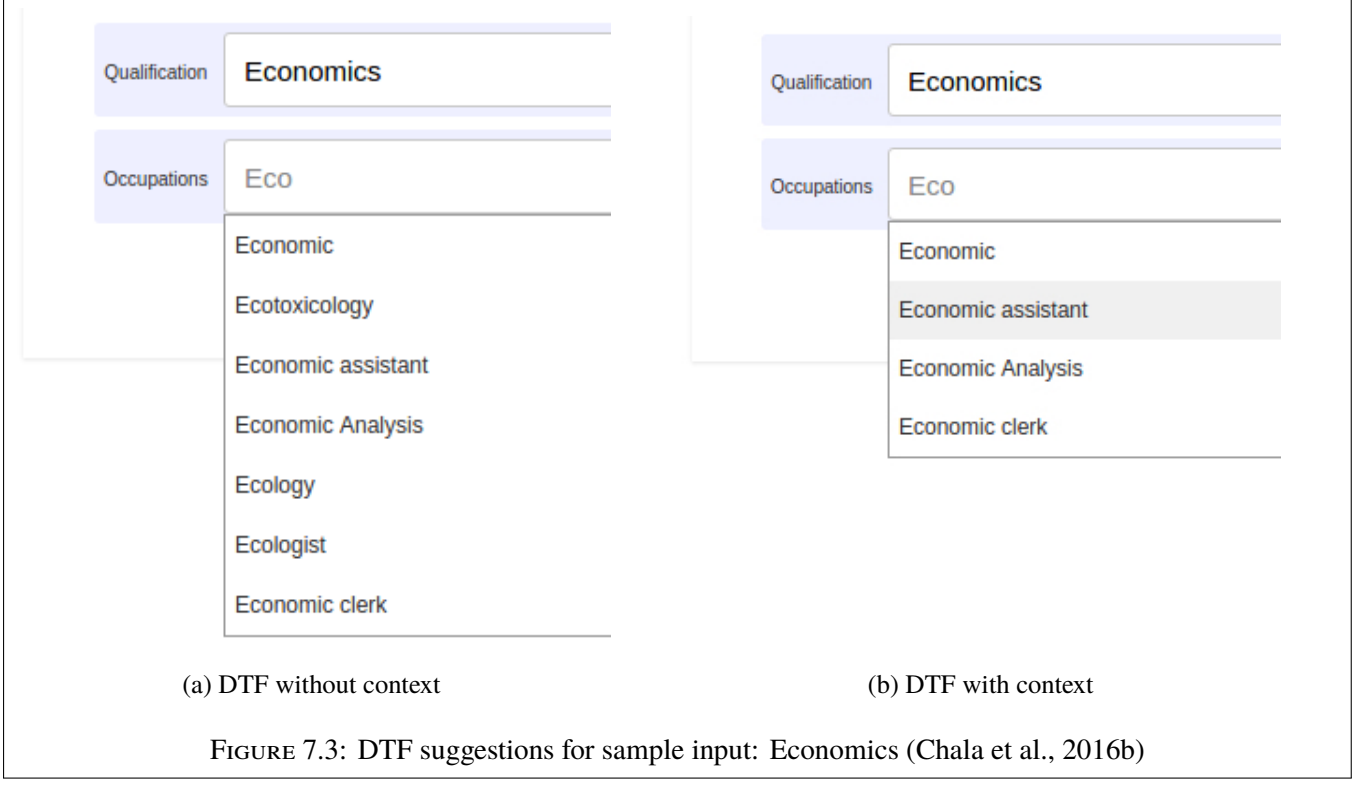


FIGURE 7.2: DTF to select qualification

Without the implementation of context-aware DTF, after the user enters one of the qualifications, elements in all occupations will be considered to be filled in the list of suggestions irrespective of whether they are semantically relevant to the qualification or not as shown in Figure 7.3a and 7.4a. From Figure 7.3a, which performs autocompletion without the DTF, we can observe that the content that may be interesting to the user are *Economic*, *Economic assistant*, *Economic Analysis* and *Economic clerk*, whereas *Ecotoxicology*, *Ecology* and *Ecologist* are not interesting in relation to his/her entry in the *Qualification* field.

Similarly, as shown in Figure 7.4a, the user is provided with choices which may not be interesting in relation to the previous entry in *Qualification* field, i.e., *Environmental protection*. The user should not

have been given *Economic*, *Economic assistant*, *Economic Analysis* and *Economic clerk*. Because none of these entries are strongly related with the previous entry.



(a) DTF without context

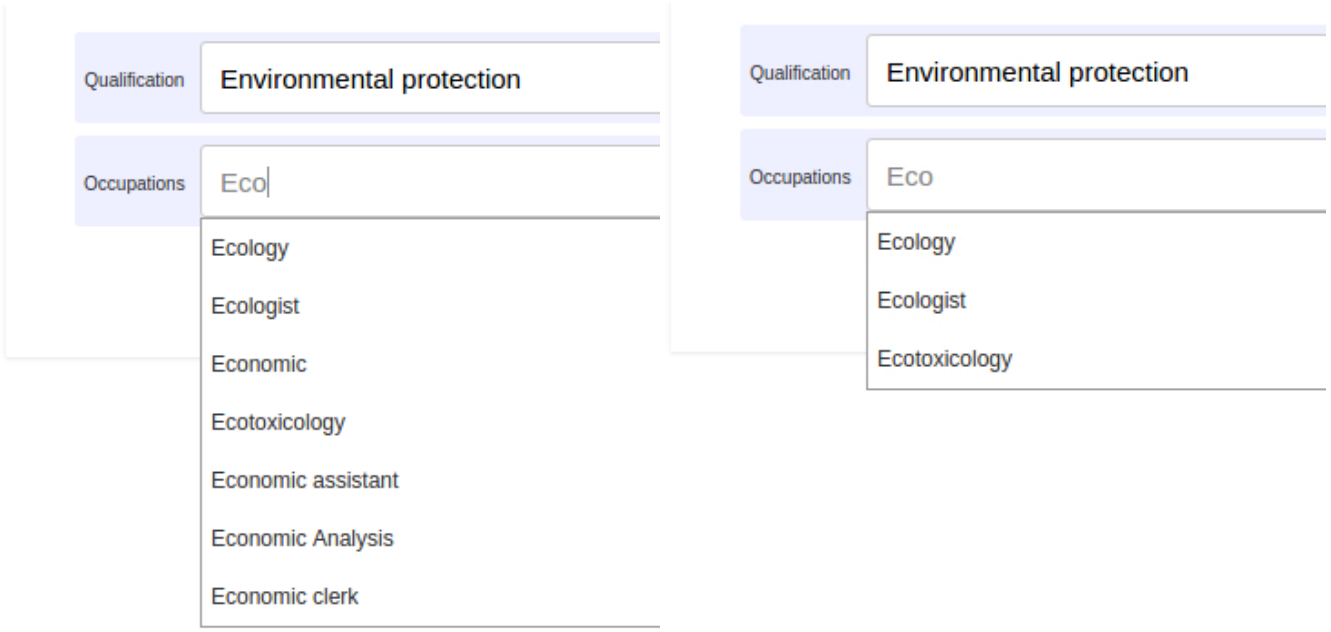
(b) DTF with context

FIGURE 7.3: DTF suggestions for sample input: Economics (Chala et al., 2016b)

In contrast, with context-aware DTF, once the user enters the qualification, occupations which are non-relevant to his/her entry are excluded from the suggestion list as shown in Figure 7.3b. For the user who has qualification in the area of *Economics* as entered in the *Qualification* field, suggestions are reduced to the most relevant ones, namely *Economic*, *Economic assistant*, *Economic Analysis* and *Economic clerk*. Likewise, if a user in the field of environmental protection starts filling data, the suggestion list is updated to contain elements in related field, namely *Ecotoxicology*, *Ecology* and *Ecologist*, as shown in 7.4b. Filtering out the data in order to provide list of suggestions that are only relevant to the job seeker, i.e., based on previous entries, offers focused and compact options. While the system provides list of suggestions that are as close to the previous entries as possible, the user is still free to keep writing anything of his/her interest when he/she thinks the suggestions offered are not among what he/she wants to enter.

The suggestions produced by DTF have been evaluated for the information quality of their content using the framework of measuring information quality developed by Eppler (2006). There are a wide range of

quality criteria in Eppler (2006) for evaluating implementation of DTF, i.e., in relation to four quality levels, namely i) the community (relevance), ii) product (soundness), iii) process and iv) infrastructure level.



(a) DTF without context

(b) DTF with context

FIGURE 7.4: DTF suggestions for sample input: Environmental Protection (Chala et al., 2016b)

Identification and application of the quality criteria, on the one hand, pertains to the end-user demands and his/her assessment of the dynamically supplied recommendations On the other hand, it pertains to the recommendation object itself. While the four levels cover the former, community and product level in particular addresses the latter, i.e. quality of recommendation object (output of the system). In this study, we only discuss the latter issue especially with regard to the relevance and soundness of the recommendations (i.e., community and product level). Table 7.1 summarizes the selected criteria to evaluate the quality of recommendation objects.

The framework discussed in (Eppler, 2006) summarized in Table 7.1 was used as a qualitative metric to measure the DTF suggestions. As shown in 7.3b and 7.4b (i.e., form entry using DTF), the results satisfy all the criteria in both relevance and soundness criteria levels. The suggestions provided are relevant, i.e., comprehensive, accurate, clear and applicable, for the given partial entry. Similarly, the entries are sound, i.e., concise, consistent, current, correct and convenient, for the given partial entry.

Criterion Level Level	**Criterion Name Name**	**Description**
Community Level (Relevance)	Comprehensiveness	Is the scope of recommendation adequate? (Completeness over the domain/topic)
	Accuracy	Is the recommendation precise enough and close enough to selected domain/topic?
	Clarity	Is the recommendation understandable or comprehensible to the target group (end-user)?
	Applicability	Can the recommendation be directly applied? Is it useful?
Product Level (Soundness)	Conciseness	Is the recommendation to the point, void of unnecessary elements?
	Consistency	Is the recommendation free of contradictions or convention breaks?
	Currency	Is the recommendation up-to-date and not obsolete?
	Correctness	Is the recommendation free of distortion, bias, or error?
	Convenience	Does the recommendation provision correspond to the user's needs and habits?

TABLE 7.1: Description of the selected criteria for evaluating the quality of recommendations – Adopted from (Eppler, 2006)

In contrast, the results of the suggestions shown in 7.3a and 7.4a (i.e., form entry without using DTF), suffers from not satisfying one of the criteria in Relevance (i.e., accuracy) and two of the criteria in Soundness (i.e., conciseness and convenience).

Given the pervasiveness of auto-completion in search engines such as in Google®, one may wonder if there is any difference between those systems and DTF. Although, DTF is similar to those systems in that they both make recommendation to assist user in data entry, there are a number of differences between them. First, the type of data used is different. The first difference is that there is a significant difference in the diversity and volume of the data they operate with. Obviously, Google' autocomplete has much bigger and far more diverse data than that used in this research. That is, the dataset used here is specific to occupation and vacancy while that of Google is virtually everything. As a result, the precision of the suggestion will also be different. The recommendation of DTF is much more focused to the domain of occupation than that of Google's and thus it is more precise in this domain than Google's is whereas it

does not do anything in other domains where Google's autocomplete does something.

Secondly, source of data input used is different. The most important difference is that the recommendation of Google is entirely based on previous queries than the underlying data, whereas, the recommendation of DTF is based on the underlying data that is crawled from occupational standardization systems and vacancies. Google stores previous queries by other users and recommends the most likely (the most frequently entered) query to user's partial input. As a result, it sometimes makes unrelated recommendations. This occurs not because of the underlying data but because of the previous queries. As stated above, basing its analysis of the underlying data, DTF recommends better to queries in the specified domain.

The third difference is that in the event that the user does not select any of the suggestions, Google saves the users entry, as queries to use it in subsequent queries whereas DTF saves the user's entry as more data and recomputes the distance and co-occurrence matrix for future recommendation taking into account the current input.

Fourth, there is also a difference with regards to the objectives, i.e., system goals. Google's system is intended for searching whereas DTF is intended for data collection though they both do recommend the user with list of suggestions to enter.

Finally, there may also be a difference in effect of recommendations on user. Google's recommender has a tendency to distract users' attention and may lead to higher dropout rate by suggesting unrelated but apparently interesting entries.

7.1.2 Bidirectional Candidate to Vacancy Matching

Evaluating efficiency and productivity of matching process has been studied in Hall and Schulhofer-Wohl (2015) with the number of active job seekers and vacant jobs as inputs and job finding as output. That is, matching efficiency as seen by Hall and Schulhofer-Wohl (2015) is a measure of monthly job-finding rate, focusing only on the number of active job seekers getting jobs and vacant jobs filled. Evaluation of job matching efficiency addressed in this research differs from Hall and Schulhofer-Wohl (2015) in that the former measures the level of relevance of the matching job seeker to the job whereas the latter measures the number of job findings.

In order to demonstrate that, let us consider the sample dataset[1] of job seekers in Table 4.3 and vacancies in Table 5.3. The matching result is visualized on Figures 7.5 and 7.6.

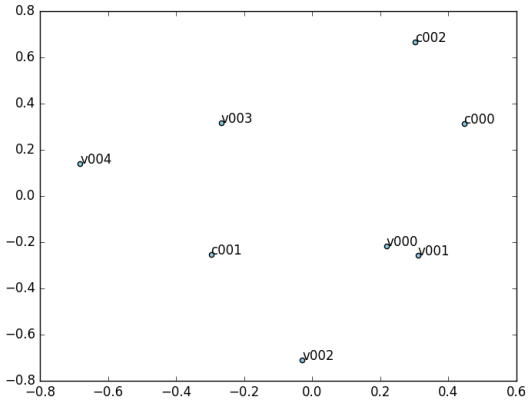


FIGURE 7.5: Matching vacancies to job seeker

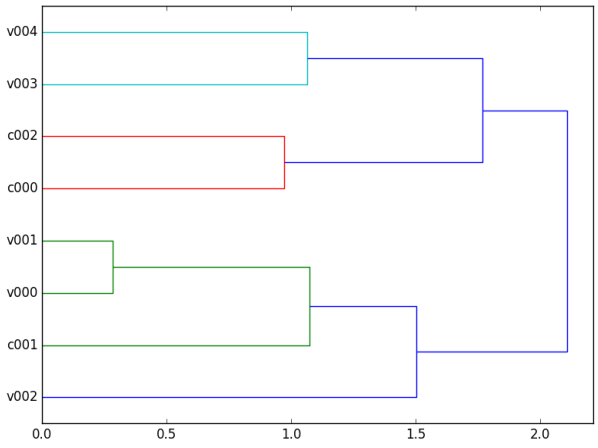


FIGURE 7.6: Matching job seekers to vacancy

As shown in Figure 7.5 and 7.6, one can see that candidate job seeker *c001* is closer to vacancies *v000*, *v001*, and *v002*. Close observation to this sample data shows candidate job seeker *c001* has the highest similarity to *v002* than any other vacancies in this sample. However, because *IOT* is considered lexically different from *Internet of Things*, the similarity was lower and hence not the top match. To address this,

[1]This is a gross over simplification for the sake of visibility. The actual data used in the study, after cleaning and preprocessing, had 181 features

during preprocessing, all occurrences of *Internet of Things* were replaced with its acronym equivalent, i.e., IoT. This has improved the undesired results such as the one shown in Figure 7.6.

7.1.3 Inclusion of Social Networking Data

Data of individuals from professional network that contains professional details of persons (i.e., attributes) and information produced due to interaction of these individuals on topics (i.e., relations) is also collected from online open access sources. After cleaning noisy data and eliminating individuals that have no data in either attributes, relations or both, 65 records of candidates with data about themselves and professional connections were obtained. From 65 job seekers to test the system, using A/B Testing (Hanington and Martin, 2012) which measures the difference in performance of the system when a certain parameter is included (as compared to the performance in the absence of the parameter). The result shown in the confusion matrix in Table 7.2 is when matching job seeker with vacancy in the absence of social networking data, whereas, the result shown in Table 7.3 is when matching job seeker with the same job vacancy using social networking data included.

		Predicted		
		Match	Not match	Total
Actual	Match	7	9	16
	Not match	4	45	49
	Total	11	54	65

TABLE 7.2: Matching without social networking data (Chala and Fathi, 2017)

		Predicted		
		Match	Not match	Total
Actual	Match	10	6	16
	Not match	3	46	49
	Total	13	52	65

TABLE 7.3: Matching with social networking data (Chala and Fathi, 2017)

Evaluation is done using accuracy measure as shown in Equation 7.1 because it is an evaluation metric that gives a single measure of quality (Manning et al., 2008). The accuracy measure which is the ratio of correct matches (i.e., the number of correctly matched job seekers) to total number of job vacancies to evaluate the effectiveness, such that *CM* is the number of individuals that are actually correct matching to the job under consideration, *CNM* is the number of individuals that have relevant skills but not matched, *NCM* is the number of individuals that do not have relevant skills but matched, and *NCNM* is the number of individuals that do not have relevant skills and are not matched. Then,

$$accuracy = \frac{CM + NCNM}{CM + CNM + NCM + NCNM + \delta} \tag{7.1}$$

where, δ stands for the number of distinct vacancies less the sum of *CM, CNM, NCM, NCNM,* i.e., the number of vacancies that did not show up in the confusion matrices (cf. Table 7.2 and Table 7.3) which are excluded during clustering phase as described in the bidirectional matching algorithm. The experiment shows promising result in achieving accuracy improvement. The number of incorrect matches are reduced (from 4 to 3) while the number of correct matches are increased (from 7 to 10). As shown in Table 7.2 and Table 7.3, the percentage accuracy increased from 47.27% to 50.91% in matching job seekers to job vacancies due to addition of social network data. Thus, inclusion of social network data in skill measurement shows consistent improvement in accuracy, as shown in Figure 7.7, where the blue bars show the matching with social networking data included, while the red bars show the matching without social networking data.

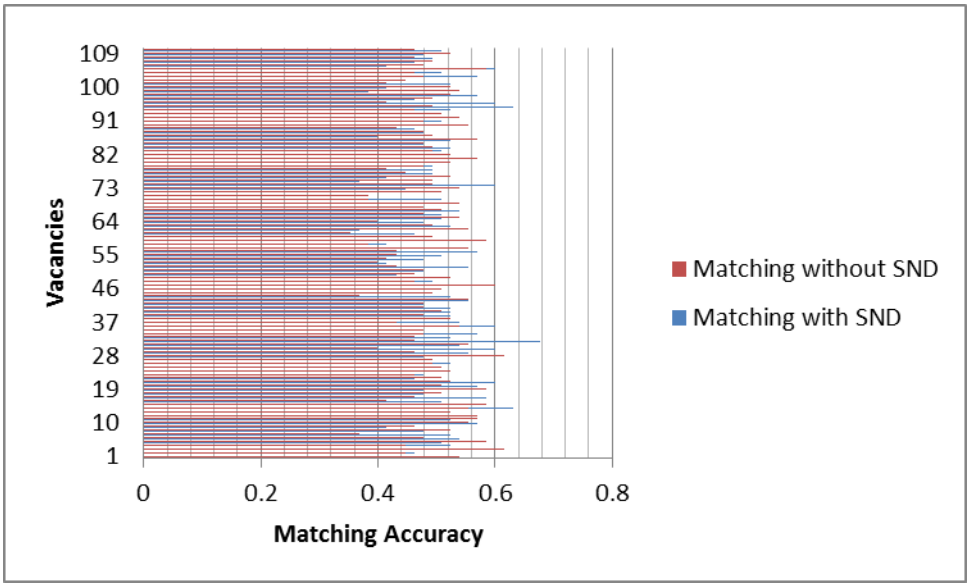


FIGURE 7.7: Matching accuracy with and without social network data (SND) (Chala and Fathi, 2017)

7.2 Contributions

This research contributes to the current body of knowledge in job matching by applying data-intensive methods in job seeker and vacancy modeling and matching. On one hand, it involves job seeker profile enriching using online self-assessment survey data, implementing DTF-enabled context-aware user assistance on job seeker profile survey (Chala et al., 2016b), weighting job seeker profile with social networking data (Chala and Fathi, 2017), and preference of a job seeker obtained through web survey. On the other hand, it involves enriching job vacancy using occupational standard data (Chala et al., 2016a) in order to model vacancies for better matching.

Enriching Vacancies using standard occupations not only provided more accurate and up-to-date information that is useful for job designers to develop more specific job descriptions. It also supports employers or job-agents to identify crucial and cross-cutting skill sets to be stated in the requirements for a vacancy advertisements. Through enriching vacancies with occupational standards, missing features in vacancy advertisement are filled and existing features will have their weights boosted.

With the goal of improving user experience, the system provides numerous options to collect inputs from the job seeker. It collects user entries on DTF-enhanced web interface to add to the data from job seeker resumé via content extraction and transforming the data to suit the matching and modeling system. The context-aware DTF adjusts the user interface elements' responses to user interaction depending on the

user's domain of qualification. On condition of entry of one variable, e.g., qualification, which will use the global data as a suggestion pool, the next input fields are adjusted not only by the current partial input but also on the previously entered full text. This reduces search space thereby improving system resource utilization, i.e., it reduces response time, and avoids unrelated items from the suggestion list.

Integrating social networking data into the equation of job matching improved the job seeker matching to vacancies. With social networking data, job seeker features are either expanded when they are absent in data from the web survey and resumé, or their weight in the feature vector is increased so that the feature gets more weight in the matching process. By doing so, this contributes to the current literature with a prototype job matching system that integrates social networking data, as part of job seeker modeling system for vacancy matching and recommendation.

Job matching data generated online is expensive to label and performing natural language processing is inefficient on the raw data. However learning features from unstructured text to building a document vector in order to represent them using statistically most important words contained in the document using unsupervised learning is proved to perform well. Hence, deep learning is applied in this research for vacancy and job seeker matching and found a promising result in performance and accuracy.

Chapter 8

Conclusion and Future Works

This chapter concludes the dissertation by summarizing the research problems, objectives, methods utilized and the results achieved. It also covers the implication of this study for the users, i.e., job seekers and employers, and its applications in human resource development processes other than recruitment. By discussing the assumptions and limitations of the study, the chapter highlights areas of future research that extends job matching.

8.1 Conclusion

Bidirectional job matching is the process of selecting a candidate job seeker that satisfies requirements of a vacancy, and vice versa, using semantic similarity of the two entities.

Matching job seekers to vacancies has been a challenging task in recruitment industry. The challenges are due to difficulties in processing and understanding job seekers, vacancies, or matching process. Despite public availability of online vacancies, a large number of jobs are not filled due to lack of the right applicant. This is partly because the online vacancy data is in overwhelming volume for job seekers to find relevant vacancy for their skill.

Expansion of in the Internet access and growth in online recruitment has motivated job seekers and job holders to produce huge online jobseeker data. With this available large volume of data which is already

in electronic form, predictive analysis techniques help users to make better decisions, take more consistent actions and reduce costs.

This study demonstrated how natural language processing and machine learning can be used to match job seekers to vacancies using features extracted from textual data. After discussing the theoretical foundations and practical applications in existing literature, it extends the current body of knowledge in the field by developing a framework that utilizes machine learning techniques in job seeker to vacancy matching using multifaceted data, i.e., resumé, web survey, social networking, online vacancy advertisements, occupational standards data. After collecting the resumé and vacancy data from the web, it applies NLP in the process of job matching.

Although many of the methods employed in this research are applicable in various domains, this research focused only on the IoT jobs vacancies. It is also important to note that this study is meant to expedite the recruitment process and its accuracy of suggestions is yet to be tested by expert recruiters.

Using DTF in web-based data collection by combining capacity of text analysis with real-time client-server communication, gives myriads of benefits. First, the data entry task on the part of the user will be simpler because the guided data entry will reduce end-user side pressure that would be present when using open-ended user interface elements. It also provides more flexible and extended options for end users to choose from - a feature that would not be present in the application of a close-ended user interface element.

Second, the integration of text analysis in the system enables DTF to suggest from existing entries. This helps synthesize options which are too close to be distinct. Within systems that use TextBox, same data is stored as if it was different due to subjectivity factor, i.e., when different users enter same content in different diction. The system suggests a list of choices for the user to choose based on previous entries and without imposing on him/her to choose from the given list. This allows subjectivity by giving user freedom to enter anything. If the user realizes that what he/she is going to type is among the entries in the suggestion list, then he/she will proceed to make a choice of the best entry as per his/her intention. The ultimate effect is thus increasing objectivity by maintaining subjectivity. Employing DTF for data entry, therefore, leads to a balance between subjectivity and objectivity.

Third, when distinct entries do arise, the system learns to expand the database content by the addition of new job title, skill, description, task and/or qualification. This, in turn, will be used as input to the

text analysis module for future suggestion – and subsequently allows for continuous and incremental adaptability.

Fourth, implementing web-based data collection system using DTF is better than using either open- or close-ended user interface elements. It improves the engagement of the end user with the system by simplifying data entry. By doing so, it improves the quality of data that will be collected and its continuous cycle of learning makes it self-adaptive.

The baseline matching system depends on resumé and vacancy advertisement data to represent job seekers and vacancies, respectively. Experiments using data that represent different perspectives show that resumés and vacancy advertisements do not sufficiently represent the two concepts. Job descriptions in vacancies are often not complete, i.e., employers fail to stipulate some information in either essential or preferred requirements or both during vacancy announcement.

Enriching vacancies with occupational standards increased the inclusion of the vacancy in the recommendation by adding more features. Because vacancy advertisements lack descriptions that are present in occupational standards. Similarly, web survey and social networking data provide more aspects of a job seeker that resumé cannot provide. Utilizing social networking and web survey data in modeling job seeker showed improved matching, as compared to using only resumé.

8.2 Implications

The system has benefits/values for different users: i) it helps employers in matching employees to vacancy, employee assessment for promotion, and assisting in preparation of vacancies; ii) it helps job seekers in matching vacancies to job seekers; iii) it helps recruitment agents in efficient matching of job seekers to vacancies and vice versa.

The system can be applied in internal mobility of employees, within an organization, i.e., employee promotion, taking into account employee information stored in the human resource department such as salary information, service period, supervisor evaluation, etc. Using skill analysis for staff promotion is interesting use case to apply the system internally within an organization matching to decide whether an employee is eligible to be promoted.

At national level, it can also have macroeconomic benefit by helping facilitate the labour employability. By improving matching, unfilled vacancies will be filled by job seekers. When job seekers are placed in the unfilled vacancies, there will be fewer unemployed people in the labour force. This can lead to reduction of government welfare costs that arise due to unemployment.

8.3 Assumptions and Limitations

In addition to data obtained from the project partners, a large volume of vacancy advertisement data was collected from various open access Internet sources, and analyzed to represent and model vacancy. Similarly, a large volume of resumé was collected from various open access Internet sources, and analyzed to represent and model job seeker. Integrating data collected about job seekers, i.e., resumé, social networking, and web survey data is performed in order that a unified measure of a job seeker feature is computed. However this data does not include transaction data, i.e., actual job seeking information of job seekers.

Because of this limitation, job seekers and vacancies are clustered using standard occupation title as intermediate variable that connects the two concepts – job seeker and vacancy, i.e., job seekers are matched to occupational title. Likewise, vacancies are mapped to occupational titles. The clustering assumption is that each concept that maps to same occupational title belongs to one entity on which bidirectional matching and ranking is performed.

Because the recommendations are too specific, they may not let some job seekers who aspire for career switch as they do not provide other closely related vacancies. Thus the system only effectively supports those job seekers who have the requisite background and are aspiring to careers that match their already established profile. If users want demand-driven career switch plan, this system may not satisfy their need.

8.4 Future Research

There are a number of areas for future works. The future works in this research include rethinking data preprocessing, interactive mobile communication for job vacancy recommendation, and labeling of

data via expert validation for use with supervised machine learning methods that might further improve matching job seekers with vacancies.

Rethinking role of stopwords through chunking, i.e., grouping connected items or words together so that they can be processed or stored as single term and keeping them, in feature extraction for text matching is important. Because removing stopwords during data cleaning phase undoubtedly results in loss of useful context information (Miner et al., 2012). However in, eliminating bi-grams that are made up of two stopwords are eliminated without affecting the performance of matching and contributing much in dimensionality reduction. For example, eliminating "of the", "to the", and "that has" does not have same effect in information lose as "to work" although some bi-grams have specific meanings such as "of the" showing possession.

Because of the ubiquity of mobile devices, their ease of use, and availability of cloud system for process- and memory-intensive tasks, implementing mobile job vacancy recommendation system is of paramount importance to capture useful information from job seeker. When a new idea flashes into the mind of a job seeker, he/she will be able to easily provide information over mobile application. This continuous update of the job seeker information enables personalized job recommender system to fine-tune the results. Moreover, using mobile based system, we can also capture more information about the job seeker such as location and language preference automatically to assist in the data entry.

It is also important to integrate of feedback loop in order to incorporate users' opinion as to how good the matching performed in satisfying their needs. The level of improvement in user engagement as well as data quality by reducing dropout rates needs to be quantified through conducting research on users (for instance by piloting the job matching implementation prototype).

Through interaction with user, the parameters need to be tuned to user's preferences. Then the parameters need to be weighted according to their importance, which in turn is dependent on the personal preferences of the job seeker.

The job seekers also give other preferences (that are non-skill) that can affect their response to take on a vacancy. For example, the job seeker may prefer one workplace over another or likes one job more than another. The system needs to allow the job seekers to prioritize their skills and preferences in a scale. These priorities need to change the weights that are used to tell what skills describe the job seeker most.

Due to the labor force mobility especially across Europe (Fischer et al., 2014) and individual-skill mismatches (Carnevale et al., 2014, Godliman Partners, 2009), the scope of this study needs to be extended to analyze multi-lingual job descriptions and job vacancies. Integrating Cross-Lingual Information Retrieval (CLIR)(Grefenstette, 2012) techniques, the system can work for multiple languages in such a way that the user will be assisted in multiple languages in the data entry and the matching works for multiple languages.

Glossary

Artificial intelligence

is the simulation of human intelligence processes – learning (the acquisition of information and rules to utilize the information), reasoning (using the rules to derive conclusions), and self-improvement – by computer systems.

Big Data

is the term for a collection of data sets so large and complex that it becomes difficult to process using on-hand database management tools or traditional data processing applications.

Close-ended format

user interface element with strictly limited option, e.g., checkbox, radio button, list box.

Cloud Computing

is a term used to describe the practice of using computing and software resources that are available on remote servers connected to the Internet and are delivered on demand, as service to store, manage, and process data.

Clustering

Clustering is a set of algorithms that perform analysis of very large data sets in order to partition items into groups based on their similarity.

Data analysis

unlike data mining, data analysis starts with a specific hypothesis and applies machine learning to analyze (big) data with the objective of testing the hypothesis.

Data management

refers to a scalable, efficient and reliable ways to store, manage and process (un)structured data.

Data mining

refers to a process that uses machine learning as a tool in its search for new knowledge from big data without any preconceived notion or hypothesis.

knowledge base

is a dynamic storage of facts and inference that has the capacity to update itself through learning by artificial intelligence..

Knowledge discovery

describes the process of automatically searching large volumes of data for patterns that can be considered knowledge about the data.

Knowledge management

refers to a range of practices used by organizations to identify, create, represent, and distribute knowledge for reuse, awareness and learning across the organization.

Machine Learning

system of building computer programs that automatically improve with data.

O*NET

Occupational Information Network - an occupational information system by the US Department of Labor.

Occupation

a regular activity performed for payment, that occupies one's time.

Ontology

is a term used to representation of knowledge as a hierarchy of concepts within a domain, using a shared vocabulary to denote the types, properties and interrelationships of those concepts.

Open-ended format

user interface element with free-form entry, e.g., text box, text area.

Spider

a softbot that trawls web pages to scrape content from them.

Stopwords

Words that occur so frequently that their presence or absence does not affect the meaning of a document.

Underemployment

the situation in which people in a labor force are employed at in jobs with less than full-time or at jobs inadequate with respect to their training.

Unemployment

the situation in which an active job seeker is not having a job. It also refers to proportion of unemployed people to the total workforce.

Web mining

collection, extraction and integration of web data, process it and make useful information or knowledge out of it.

Bibliography

Academia.edu (2016). Academia.edu share research. Available at: http://www.academia.edu.

Aciar, S., Zhang, D., Simoff, S., and Debenham, J. (2007). Informed recommender: Basing recommendations on consumer product reviews. *IEEE Intelligent Systems*, 22(3):39–47.

Alba, D. (2015). AI software that could score you the perfect job.

Albert, W. and Tullis, T. (2013). *Measuring the user experience: collecting, analyzing, and presenting usability metrics*. Newnes.

Amin, A. K., Hildebrand, M., van Ossenbruggen, J. R., Evers, V., and Hardman, L. (2009). List, Group or Menu: Organizing Suggestions in Autocompletion Interfaces. CWI.

Anderson, J. R. (2000). *Learning and memory*. John Wiley New York, NY.

Arcaute, E. and Vassilvitskii, S. (2009). Social networks and stable matchings in the job market. In *International Workshop on Internet and Network Economics*, pages 220–231. Springer.

Athavaley, A. (2007). Job references you can't control. *The Wall Street Journal*, 27.

Ayodele, T. O. (2010). *Types of machine learning algorithms*. INTECH Open Access Publisher.

Back, A. and Koch, M. (2011). Broadening participation in knowledge management in enterprise 2.0. *it-Information Technology Methoden und innovative Anwendungen der Informatik und Informationstechnik*, 53(3):135–141.

Baskerville, R. and Dulipovici, A. (2015). The theoretical foundations of knowledge management. In *The Essentials of Knowledge Management*, pages 47–91. Springer.

Beansprock (2017). Beansprock: Your best job delivered. Available at: https://www.beansprock.com/user/profile.

Bektas, E. (2013). *Knowledge sharing strategies for large complex building projects*. TU Delft.

Belloni, M., Brugiavini, A., Meschi, E., and Tijdens, K. G. (2016). Measurement error in occupational coding: an analysis on share data. *Journal of Official Statistics*, 32(4):917–945.

Bengio, Y. et al. (2009). Learning deep architectures for AI. *Foundations and trends® in Machine Learning*, 2(1):1–127.

Bessen, J. (2014). Employers aren't just whining: The 'skills gap' is real. *Harvard Business Review*, 25.

Beukes, R., Fransman, T., Murozvi, S., Yu, D., et al. (2016). Underemployment in south africa. Technical report.

Bhat, Z. H. (2014). Job matching: the key to performance. *International Journal of Research in Organizational Behavior and Human Resource Management*, 2(4):257–269.

Bhavsar, H. and Ganatra, A. (2012). A comparative study of training algorithms for supervised machine learning. *International Journal of Soft Computing and Engineering (IJSCE)*, 2(4):2231–2307.

Biemann, C. (2012). *Structure Discovery in Natural Language*. Springer.

Boschetti, A. and Massaron, L. (2016). *Python data science essentials*. Packt Publishing Ltd.

Bottou, L. (2012). Stochastic gradient descent tricks. In *Neural networks: Tricks of the trade*, pages 421–436. Springer.

Branham, L. (2012). *The 7 hidden reasons employees leave: How to recognize the subtle signs and act before it's too late*. AMACOM Div American Mgmt Assn.

Brown, M., Setren, E., and Topa, G. (2016). Do informal referrals lead to better matches? evidence from a firm's employee referral system. *Journal of Labor Economics*, 34(1):161–209.

Bruch, M., Monperrus, M., and Mezini, M. (2009). Learning from Examples to Improve Code Completion Systems. In *Proceedings of the the 7th joint meeting of the European software engineering conference and the ACM SIGSOFT symposium on The foundations of software engineering*, pages 213–222. ACM.

BusinessDictionary.com (2017). Knowledge base. Available at: http://www.businessdictionary.com/definition/knowledg.

Cahuc, P. and Fontaine, F. (2002). On the efficiency of job search with social networks.

Callan, V. J. and Bowman, K. (2015). Industry restructuring and job loss: helping older workers get back into employment.

Cantador, I., Bellogín, A., and Castells, P. (2008). News@ hand: A semantic web approach to recommending news. In *Adaptive Hypermedia and Adaptive Web-Based Systems*, pages 279–283. Springer.

Capiluppi, A. and Baravalle, A. (2010). Matching demand and offer in on-line provision: A longitudinal study of monster.com.

Cappellari, L. and Tatsiramos, K. (2015). With a little help from my friends? quality of social networks, job finding and job match quality. *European Economic Review*, 78:55–75.

Career Evolutions (2016). C++ programmer maynard, ma.

Carnevale, A. P., Jayasundera, T., and Repnikov, D. (2014). The online college labor market: Where the jobs are. *Georgetown University Center on Education and the Workforce*.

CDC (2016). CDC - Industry and Occupation Coding: NIOCCS System Overview.

Cedefop (2008). Skill needs in europe: Focus on 2020.

Cedefop (2009). The dynamics of qualifications: defining and renewing occupational and educational standards. Luxembourg: Office for Official Publications of the European Communities.

Cedefop (2014). Skill mismatch: more than meets the eye.

Cedefop (2015). Matching skills and jobs in europe: insights from cedefop's european skills and jobs survey.

Celebi, M. E. and Aydin, K. (2016). *Unsupervised Learning Algorithms*. Springer.

Chala, S., Ansari, F., and Fathi, M. (2016a). A framework for enriching job vacancies and job descriptions through bidirectional matching. In *12th International Conference on Web Information Systems and*

Technologies, volume 2, pages 219–226. SCITEPRESS Digital Library (Science and Technology Publications, Lda).

Chala, S. and Fathi, M. (2017). Job seeker to vacancy matching using social network analysis.

Chala, S. A., Ansari, F., and Fathi, M. (2016b). Towards implementing context-aware dynamic text field for web-based data collection. *International Journal of Human Factors and Ergonomics*, 4(2):93–111.

Charu, C. A. and Zhai, C. X. (2012). *Mining Text Data*. Springer.

Chen, K., Hellerstein, J. M., and Parikh, T. S. (2010). Designing Adaptive Feedback for Improving Data Entry Accuracy. In *Proceedings of the 23nd annual ACM symposium on User interface software and technology*, pages 239–248. ACM.

Christen, P. (2012). *Data matching: concepts and techniques for record linkage, entity resolution, and duplicate detection*. Springer Science & Business Media.

Coates, A., Ng, A., and Lee, H. (2011). An analysis of single-layer networks in unsupervised feature learning. pages 215–223.

Couper, M. P. and Miller, P. V. (2008). Web Survey Methods Introduction. *Public Opinion Quarterly*, 72(5):831–835. AAPOR.

Coutsoukis, P. (2011). Dictionary of Occupational Titles DOT - Job Descriptions.

Dash, M., Liu, H., and Yao, J. (1997). Dimensionality reduction of unsupervised data. In *Tools with Artificial Intelligence, 1997. Proceedings., Ninth IEEE International Conference on*, pages 532–539. IEEE.

Dayan, P., Sahani, M., and Deback, G. (1999). Unsupervised learning. *The MIT encyclopedia of the cognitive sciences*.

Deng, L., Yu, D., et al. (2014). Deep learning: methods and applications. *Foundations and Trends® in Signal Processing*, 7(3–4):197–387.

Deveria, A. (2016). Can I Use... Support tables for HTML5, CSS3, etc.

Dillman, D. A., Smyth, J. D., and Melani, L. (2011). *Internet, Mail and Mixed-Mode Surveys. The Tailored Design Method*. JSTOR.

Doan, A., Naughton, J. F., Ramakrishnan, R., Baid, A., Chai, X., Chen, F., Chen, T., Chu, E., DeRose, P., Gao, B., et al. (2009). Information extraction challenges in managing unstructured data. *ACM SIGMOD Record*, 37(4):14–20.

Domeniconi, G., Moro, G., Pasolini, R., and Sartori, C. (2015). A comparison of term weighting schemes for text classification and sentiment analysis with a supervised variant of tf. idf. In *International Conference on Data Management Technologies and Applications*, pages 39–58. Springer.

Dong, X. L. and Srivastava, D. (2013). Big data integration. In *Data Engineering (ICDE), 2013 IEEE 29th International Conference on*, pages 1245–1248. IEEE.

Dos Santos, C. N. and Gatti, M. (2014). Deep convolutional neural networks for sentiment analysis of short texts. In *COLING*, pages 69–78.

Dumoulin, V. and Visin, F. (2016). A guide to convolution arithmetic for deep learning. *arXiv preprint arXiv:1603.07285*.

Dunham Group Inc (2017). Systems administrator. Available online at: https://thedunhamgroup.com/uploads/docs/systems-administrator.pdf.

Dunlop, J. T. (1966). Job vacancy measures and economic analysis. pages 27–47. National Bureau of Economic Research.

Eduworks (2014). Eduworks: Better matching through bigger data.

Eppler, M. J. (2001). A generic framework for information quality in knowledge-intensive processes. In *Proceedings of the Sixth International Conference on Information Quality*, pages 329–346.

Eppler, M. J. (2006). *Managing Information Quality: Increasing the Value of Information in Knowledge-intensive Products and Processes*. Springer Science & Business Media.

European Commission (2015). ESCO - European Skills, Competences, Qualifications and Occupations.

European Commission (2017). Esco occupations - systems administrators.

European Commission and ECORYS (2012). European vacancy and recruitment report 2012.

Evans, J. R. and Mathur, A. (2005). The Value of Online Surveys. *Internet Research*, 15(2):195–219. Emerald Group Publishing Limited.

Facebook (2016). Facebook. Available at: http://www.facebook.com.

Fasulo, D. (1999). An analysis of recent works on clustering algorithms.

Ferrara, E., De Meo, P., Fiumara, G., and Baumgartner, R. (2014). Web data extraction, applications and techniques: A survey. *Knowledge-based systems*, 70:301–323.

Fink, G. A. (2014). *Markov models for pattern recognition: from theory to applications*. Springer Science & Business Media.

Fischer, G., Strauss, R., and Maly, R. (2014). EU Employment and Social Situation: Recent Trends in the Geographical Mobility of Workers in the EU.

Fortunato, S. (2010). Community detection in graphs. *Journal of Physics Reports 486*, pages 75–174.

Fraternali, P., Rossi, G., and Sánchez-Figueroa, F. (2010). Rich Internet Applications. *Internet Computing, IEEE*, 14(3):9–12. IEEE.

Furtmueller, E., Wilderom, C., and Tate, M. (2011). Managing recruitment and selection in the digital age: e-hrm and resumes. *Human Systems Management*, 30(4):243–259.

Gan, G., Ma, C., and Wu, J. (2007). Data clustering: Theory, algorithms, and applications. *ASA-SIAM Series on Statistics and Applied Probability*.

Garg, R. and Telang, R. (2011). To be or not to be linked on linkedin: Job search using online social networks.

Garrett, J. J. (2010). *Elements of user experience, the: user-centered design for the web and beyond*. Pearson Education.

Gibson, B. (2006). Javascript & AJAX accessibility. Available at: http://www-03.ibm.com/able/dwnlds/AJAX_Accessibility.pdf.

Girard, A., Fallery, B., and Rodhain, F. (2014). Integration of social media in recruitment: a delphi study. In *Social Media in Human Resources Management*, pages 97–120. Emerald Group Publishing Limited.

Godliman Partners (2009). How to manage headhunters for candidates. Available at: http://godlimanpartners:com/ interface/resources/How_To_Manage_Headhunters_for_Candidates.

Gomaa, W. H. and Fahmy, A. A. (2013). A survey of text similarity approaches. *International Journal of Computer Applications*, 68(13).

Google (2015). Autocomplete - Search Help. Available online at: https://support.google.com/websearch/answer/106230?hl=en.

Greenberg, H. M. (2010). Job matching to hire motivated employees. Available at: http://precast.org/2010/05/job-matching-to-hire-motivated-employees/.

Greenwood, A. M. (1999). International definitions and prospects of underemployment statistics. *Proceedings for the Seminario sobre Subempleo, Bogota*, pages 8–12.

Grefenstette, G. (2012). *Cross-language information retrieval*, volume 2. Springer Science & Business Media.

Guarino, N., Oberle, D., and Staab, S. (2009). What is an ontology? In *Handbook on ontologies*, pages 1–17. Springer.

Guion, R. M. (2011). *Assessment, measurement, and prediction for personnel decisions*. Taylor & Francis.

Hall, R. E. and Schulhofer-Wohl, S. (2015). Measuring job-finding rates and matching efficiency with heterogeneous jobseekers. Technical report, National Bureau of Economic Research.

Hanington, B. and Martin, B. (2012). Universal methods of design: 100 ways to research complex problems. *Develop Innovative Ideas, and Design Effective Solutions: Rockport Publishers*.

Heath, T. and Bizer, C. (2011). Linked data: Evolving the web into a global data space. *Synthesis lectures on the semantic web: theory and technology*, 1(1):1–136.

Heathfield, S. (2016). Assess job fit when you select employees. Available online at: https://www.thebalance.com/assess-job-fit-when-you-select-employees-1918165.

Heckman, J. J. and Kautz, T. (2013). Fostering and measuring skills: Interventions that improve character and cognition. Technical report, National Bureau of Economic Research.

Hiermann, W. and Höfferer, M. (2003). A practical knowledge-based approach to skill management and personal development. *J. UCS*, 9(12):1398–1409.

Hillier, M. (2002). Multilingual website usability: Cultural context. In *4th International Conference on Electronic Commerce*, volume 400, pages 1–14.

Hislop, D. (2013). *Knowledge management in organizations: A critical introduction*. Oxford University Press.

Hortonworks Inc. (2014). Data science workshop. Available at: https://www.slideshare.net/hortonworks/data-science-workshop.

Hotho, A. Maedche, A. and Staab, S. (2002). Text clustering based on good aggregations. *Künstliche Intelligenz (KI)*, 16(4):48–54.

Human Capital Research Corporation (2017). Research analyst human capital research corporation. Available at: https://datajobs.com/Human-Capital-Research-Corporation/Research-Analyst-Job 8868.

Hussain, I., Hassan Kazmi, S. Z., Ali Khan, I., and Mehmood, R. (2013). Improved bidirectional exact pattern matching. *Internation Journal of Scientific and Engineering Research*.

ILO (1997). ISCO - International Standard Classification of Occupations.

Indeed.com (2017). Iot job trends. Available at: https://www.indeed.com/jobtrends/q-IoT.html.

International Confederation of Private Employment Services (2016). Employment & recruitment industry in 2014-2015.

Ioannides, Y. M. and Datcher Loury, L. (2004). Job information networks, neighborhood effects, and inequality. *Journal of economic literature*, 42(4):1056–1093.

Irishjobs (2017). IoT Jobs in Ireland, Dublin, Cork and Galway. Available at: http://www.irishjobs.ie/ShowResults.aspx?Keywords=IOT.

Itsyourskills (2016). Sample individual skills profile, individual skills profile sample.

Jansen, K. J., Corley, K. G., and Jansen, B. J. (2007). E-survey Methodology. *Handbook of Research on Electronic Surveys and Measurements*, pages 1–8. Idea Group Hershey (PA).

Jones, R. and Elias, P. (2005). CASCOT: Computer-assisted structured coding tool. *Warwick Institute for Employment Research*.

Jurafsky, D. and Martin, J. (2009). *Speech and Language Processing: An Introduction to Natural Language Processing, Speech Recognition, and Computational Linguistics*. 2nd edition edition. Prentice-Hall.

Jurafsky, D. and Martin, J. H. (2014). *Speech and language processing*, volume 3. Pearson London.

Kahana, M. J. and Adler, M. (2002). Note on the power law of forgetting. *University of Pennsylvania, unpublished note*.

Kazan, M. (2012). Underemployment: Implications for organizations. *Economic Research Institute. Redmond: Economic Research Institute*.

Keep, E. and James, S. (2010). Recruitment and selection – the great neglected topic. *SKOPE Research Paper No. 88*.

Kennan, M. A., Cole, F., Willard, P., Wilson, C., and Marion, L. (2006). Changing workplace demands: What job ads tell us. *Aslib Proceedings*, 58(3):179–96.

Khobreh, M., Ansari, F., Dornhöfer, M., and Fathi, M. (2013). An ontology-based recommender system to support nursing education and training. pages 237–244.

Khoussainova, N., Kwon, Y., Balazinska, M., and Suciu, D. (2010). SnipSuggest: Context-aware Autocompletion for SQL. *Proceedings of the VLDB Endowment*, 4(1):22–33. VLDB Endowment.

Klahold, A., Uhr, P., Ansari, F., and Fathi, M. (2014). Using Word Association to Detect Multitopic Structures in Text Documents. *IEEE Intelligent Systems*, 29(5):40–46.

Kluge, A. (2014). *The acquisition of knowledge and skills for taskwork and teamwork to control complex technical systems: A cognitive and macroergonomics perspective*. Springer.

Kochenberger, G., Glover, F., Alidaee, B., and Wang, H. (2005). Clustering of microarray data via clique partitioning. *Journal of Combinatorial Optimization*, pages 77–92.

Krishnamurthy, B. and Wills, C. E. (2009). On the leakage of personally identifiable information via online social networks. In *Proceedings of the 2Nd ACM Workshop on Online Social Networks*, WOSN '09, pages 7–12, New York, NY, USA. ACM.

Kureková, L. M., Beblavý, M., and Thum-Thysen, A. (2015). Using online vacancies and web surveys to analyse the labour market: a methodological inquiry. *IZA Journal of Labor Economics*, 4(1):18.

Kureková, L., Beblavý, M., and Thum, A.-E. (2013). Online job vacancy data as a source for micro-level analysis of employers' preferences. a methodological enquiry.

Kureková, L. M., Beblavý, M., and Thum, A.-E. (2014). Using internet data to analyse the labour market: A methodological inquiry.

Landers, R. N., Brusso, R. C., Cavanaugh, K. J., and Collmus, A. B. (2016). A primer on theory-driven web scraping: Automatic extraction of big data from the internet for use in psychological research. *Psychological methods*, 21(4):475.

Längkvist, M., Karlsson, L., and Loutfi, A. (2014). A review of unsupervised feature learning and deep learning for time-series modeling. *Pattern Recognition Letters*, 42:11–24.

Latkin, C. A., Davey-Rothwell, M. A., Knowlton, A. R., Alexander, K. A., Williams, C. T., and Boodram, B. (2013). Social network approaches to recruitment, hiv prevention, medical care, and medication adherence. *Journal of Acquired Immune Deficiency Syndromes*, 63(01):S54–S58.

Lauer, C., McLeod, M., and Blythe, S. (2013). Online Survey Design and Development: A Janus-faced Approach. *Written Communication*, 30(3):330–357. SAGE Publications.

Levenshtein, V. I. (1966). Binary codes capable of correcting deletions, insertions, and reversals. *Soviet Physics Doklady*, 10(8):707–710.

Levitin, D. J. (2014). *The organized mind: Thinking straight in the age of information overload*. Penguin.

Li, X. (1990). Parallel algorithms for hierarchical clustering and clustering validity. *IEEE Transaction on Pattern Analysis and Machine Intelligence*, page 1088–1092.

Li, Y., Chen, C.-Y., and Wasserman, W. W. (2016). Deep feature selection: theory and application to identify enhancers and promoters. *Journal of Computational Biology*, 23(5):322–336.

Lin, J. and Dyer, C. (2010). *Data-intensive Text Processing with MapReduce*. Synthesis Lectures on Human Language Technologies. Morgan & Claypool.

LinkedIn (2016). Linkedin. Available at: http://www.linkedin.com.

Liu, P., Qiu, X., and Huang, X. (2015). Learning context-sensitive word embeddings with neural tensor skip-gram model. In *IJCAI*, pages 1284–1290.

Lops, P., De Gemmis, M., and Semeraro, G. (2011). Content-based recommender systems: State of the art and trends. pages 73–105.

Lupu, M., Salampasis, M., and Hanbury, A. (2014). Domain specific search. In *Professional Search in the Modern World*, pages 96–117. Springer.

Ma, H.-D. (2011). Internet of things: Objectives and scientific challenges. *Journal of Computer science and Technology*, 26(6):919–924.

Ma, J., Xu, W., Sun, Y., Turban, E., Wang, S., and Liu, O. (2012). An ontology-based text-mining method to cluster proposals for research project selection. *IEEE transactions on systems, man, and cybernetics–part a: systems and humans*, 42(3):784–790.

Mahemoff, M. (2006). O'Reilly®Ajax Design Patterns (1st Ed). O'Reilly Media, Inc.

Maier, R. (2007). *Knowledge Management Systems: Information and Communication Technologies for Knowledge Management*. Springer, 3rd edition edition.

Mang, C. (2012). Online job search and matching quality. Available at: ftp://ftp.zew.de/pub/zewdocs/veranstaltungen/ICT2012/Papers/Mang.pdf.

Manning, C. D., Raghavan, P., and Schütze, H. (2008). *Introduction to Information Retrieval*. Cambridge University Press.

ManpowerGroup (2016). 2015 talent shortage survey. http://manpowergroup.com/talent-shortage-2015.

Manroop, L. and Richardson, J. (2016). Job search: A multidisciplinary review and research agenda. *International Journal of Management Reviews*, 18(2):206–227.

Mariani, M. (1999). Replace with a database: Oˆ* net replaces the dictionary of occupational titles. *Occupational outlook quarterly*, 43:2–9.

Marinescu, I. and Rathelot, R. (2016). Mismatch unemployment and the geography of job search. Technical report, National Bureau of Economic Research.

Markman, A. B. (2013). *Knowledge representation*. Psychology Press.

Matani, D. (2011). An $O(klogn)$ algorithm for prefix based ranked autocomplete. Citeseer.

MathWorks (2017). Machine learning technique for finding hidden patterns or intrinsic structures in data.

Mayer, A. (2011). Quantifying the effects of job matching through social networks. *Journal of Applied Economics*, 14(1):35–59.

McCormick, C. (2016). Word2vec tutorial - the skip-gram model. Available at: http://www.mccormickml.com.

McGill, T. and Dixon, M. (2013). An investigation of the impact of recertification requirements on recertification decisions. In *Proceedings of the 2013 annual conference on Computers and people research*, pages 79–86. ACM.

McKee-Ryan, F. M. and Harvey, J. (2011). "i have a job, but...": A review of underemployment. *Journal of Management*, 37(4):962–996.

McKinsey Global Institute (2014). Big data: The next frontier for innovation, competition, and productivity | McKinsey & company.

Melanthiou, Y., Pavlou, F., and Constantinou, E. (2015). The use of social network sites as an e-recruitment tool. *Journal of Transnational Management*, 20(11):31–49.

Menzies, T., Williams, L., and Zimmermann, T. (2016). *Perspectives on Data Science for Software Engineering*. Morgan Kaufmann.

Merriam-Webster (2016). Definition of matching. Available at: https://www.merriam-webster.com/dictionary/matching.

Mikolov, T., Chen, K., Corrado, G., Dean, J., Sutskever, L., and Zweig, G. (2014). word2vec.

Miner, G., Elder IV, J., and Hill, T. (2012). *Practical text mining and statistical analysis for non-structured text data applications*. Academic Press.

Mitchell, T. (1997). *Machine Learning*. McGraw Hill.

Mohri, M., Rostamizadeh, A., and Talwalkar, A. (2012). *Foundations of machine learning*. MIT press.

Mooney, R. J. and Bunescu, R. (2005). Mining knowledge from text using information extraction. *ACM SIGKDD explorations newsletter*, 7(1):3–10.

Morgan, R. L. (2008). Job matching: Development and evaluation of a web-based instrument to assess degree of match among employment preferences. *Journal of Vocational Rehabilitation*, 29(1):29–38.

Mytna Kurekova, L., Beblavy, M., and Thum, A.-E. (2014). Using internet data to analyse the labour market: A methodological enquiry.

Neffke, F. and Henning, M. (2013). Skill relatedness and firm diversification. *Strategic Management Journal*, 34(3):297–316.

Nickell, S., Nunziata, L., Ochel, W., and Quintini, G. (2003). *The Beveridge Curve, Unemployment and Wages in the OECD from the 1960s to the 1990s*. Princeton University Press Princeton.

Oracle Corporation (2013). Big data analytics – advanced analytics in oracle database. White Paper.

Otte, E. and Rousseau, R. (2002). Social network analysis: a powerful strategy, also for the information sciences. *Journal of Information Science*, 28(6):441–453.

Owen, S. and Owen, S. (2012). Mahout in action. Manning Shelter Island, NY.

Panniello, U., Tuzhilin, A., and Gorgoglione, M. (2014). Comparing context-aware recommender systems in terms of accuracy and diversity. *User Modeling and User-Adapted Interaction*, 24(2):35–65.

Paternò, F., Santoro, C., and Spano, L. D. (2009). Model-Based Design of Multi-device Interactive Applications Base on Web Services. In *Human-Computer Interaction - INTERACT 2009: 12th IFIP TC 13 International Conference*, pages 892–905. Springer.

Paulus, W. and Matthes, B. (2013). The German Classification of Occupations 2010: Structure, Coding and Conversion Table. *FDZ-Methodenreport*. Institut für Arbeitsmarkt und Berufsforschung (IAB), Nürnberg [Institute for Employment Research, Nuremberg, Germany].

Penn State University (2016). Programmer/analyst (web developer). Available online at: https://psu.jobs/job/59073.

Persch, A. C., Cleary, D. S., Rutkowski, S., Malone, H. I., Darragh, A. R., and Case-Smith, J. D. (2015). Current practices in job matching: A project search perspective on transition. *Journal of Vocational Rehabilitation*, 43(3):259–273.

Peterson, N. G., Mumford, M. D., Borman, W. C., Jeanneret, P. R., and Fleishman, E. A. E. (1999). *An Occupational Information System for the 21st Century: The Development of O*NET*. American Psychological Association.

Pilgrim, C. (2013). An Investigation of Usability Issues in AJAX Based Web Sites. In *Proceedings of the Fourteenth Australasian User Interface Conference*, volume 139, pages 101–109. Australian Computer Society, Inc.

Pizzato, L., Rej, T., Chung, T., Koprinska, I., and Kay, J. (2010). Recon: A reciprocal recommender for online dating. In *Proceedings of the Fourth ACM Conference on Recommender Systems*, RecSys '10, pages 207–214, New York, NY, USA. ACM.

Pustejovsky, J. and Boguraev, B. (1993). Lexical knowledge representation and natural language processing. *Artificial Intelligence*, 63(1-2):193–223.

Pérez-Rosés, Hebert and Sebé, Francesc and Ribó, Josep Maria (2016). Endorsement deduction and ranking in social networks. *Journal of Computer Communications*, 73:200–210.

Qian, R., Zhang, K., and Zhao, G. (2013). A topic-specific web crawler based on content and structure mining. pages 458–461.

Quintini, G. (2011). Over-qualified or under-skilled: A review of existing literature. *OECD Social, Employment, and Migration Working Papers*, (121):0_1.

Rajasekaran, S. (2005). Efficient parallel hierarchical clustering algorithms. *IEEE Transactions on Parallel and Distributed Systems*, 16(6):497–502.

Reja, U., Manfreda, K. L., Hlebec, V., and Vehovar, V. (2003). Open-ended vs. close-ended questions in web questionnaires. *Developments in Applied Statistics*, 19:159–177.

Researchgate (2016). Researchgate. Available at: http://www.researchgate.net.

Reynolds, L. and Myers, K. (2012). The incidence and persistence of youth underemployment. *Public Policy and Governance Review*, 4(2):6–15.

Russell-Rose, T. and Chamberlain, J. (2016). Searching for talent: The information retrieval challenges of recruitment professionals. *Business Information Review*, 33(1):40–48.

Şahin, A., Song, J., Topa, G., and Violante, G. L. (2014a). Mismatch unemployment. *The American Economic Review*, 104(11):3529–3564.

Şahin, A., Song, J., Topa, G., and Violante, G. L. (2014b). Mismatch unemployment. *The American Economic Review*, 104(11):3529–3564.

Santos, M. B. (2016). Beyond skill mismatch. why there are so many unfilled vacancies and simultaneously high unemployment rates?

Schaeffer, S. E. (2007). Survey: Graph clustering. *Journal of Computer Science Review*, pages 27–64.

Schleyer, T. K. and Forrest, J. L. (2000). Methods for the Design and Administration of Web-based Surveys. *Journal of the American Medical Informatics Association*, 7(4):416–425. The Oxford University Press.

Schmidhuber, J. (2015). Deep learning in neural networks: An overview. *Neural networks*, 61:85–117.

Schulz, K. U. and Mihov, S. (2002). Fast string correction with Levenshtein automata. *International Journal on Document Analysis and Recognition*, 5(1):67–85. Springer-Verlag.

Schuman, H. and Presser, S. (1979). The Open and Closed Question. *American Sociological Review*, pages 692–712. JSTOR.

Schwartz, P. M. and Solove, D. J. (2011). The pii problem: Privacy and a new concept of personally identifiable information. *NYUL rev.*, 86:1814.

Scrapit (2017). Scrapit: Web scrapping and data extraction tool. Available online at: http://scrape.it/.

Seeq Corporation (2016). Back end / full stack engineer at seeq corporation. Available online at: http://www.indeed.com/viewjob?jk=4b7a4303967e74f8.

Senthil Kumaran, V. and Sankar, A. (2013). Towards an automated system for intelligent screening of candidates for recruitment using ontology mapping (expert). *International Journal of Metadata, Semantics and Ontologies*, 8(1):56–64.

Severyn, A. and Moschitti, A. (2015). Learning to rank short text pairs with convolutional deep neural networks. In *Proceedings of the 38th International ACM SIGIR Conference on Research and Development in Information Retrieval*, pages 373–382. ACM.

Song, Q., Ni, J., and Wang, G. (2013). A fast clustering-based feature subset selection algorithm for high-dimensional data. *IEEE transactions on knowledge and data engineering*, 25(1):1–14.

Soto-Acosta, P., Soto-Acosta, P., Cegarra-Navarro, J.-G., and Cegarra-Navarro, J.-G. (2016). New icts for knowledge management in organizations. *Journal of Knowledge Management*, 20(3):417–422.

Stack Exchange Inc. (2016). Stack exchange data dump. Available online at: https://archive.org/details/stackexchange.

StepStone (2016). Your job search result. Availale at: https://www.stepstone.de/5/ergebnisliste.html?ke=Inetnet+of+things.

Sundberg, J. (2017). Top 5 job search aggregators for a smarter job hunt. Available online at: http://theundercoverrecruiter.com/top-5-job-search-aggregators-smarter-job-hunt/.

Szepesvári, C. (2010). Algorithms for reinforcement learning. *Synthesis lectures on artificial intelligence and machine learning*, 4(1):1–103.

Tait, J. I. (2014). An introduction to professional search. In *Professional search in the modern world*, pages 1–5. Springer.

Tar, H. H. and S., N. T. T. (2011). Ontology-based concept weighting for text documents. In *International Conference on Information Communication and Management, 16*.

Target Training International (2013). Job matching – the key to performance.

Text Kernel (2016). Textkernel – cv parsing, semantic search and matching software.

Thada, V. and Jaglan, V. (2013). Comparison of jaccard, dice, cosine similarity coefficient to find best fitness value for web retrieved documents using genetic algorithm. *International Journal of Innovations in Engineering and Technology*, 2(4):202–205.

Tonkin, E. (2006). AJAX And Usability Issues. http://www.ukoln.ac.uk/qa-focus/documents/briefings/briefing-94/.

Topa, G. (2011). Labor markets and referrals. In *Handbook of Social Economics*, pages 1193–1221.

Toteva, K. and Gourova, E. (2011). Social network analysis in professional e-recruitment. In *Third International Conference on Software, Services and Semantic Technologies S3T*, pages 75–80. Springer Berlin Heidelberg, Berlin, Heidelberg.

Trewin, S., Cragun, B., Swart, C., Brezin, J., and Richards, J. (2010). Accessibility challenges and tool features: An ibm web developer perspective. In *Proceedings of the 2010 International Cross Disciplinary Conference on Web Accessibility (W4A)*, W4A Í0, pages 32:1–32:10, New York, NY, USA. ACM.

Troiano, L., Cirillo, G., Birtolo, C., and Armenise, R. (2009). An application of Bayesian networks in predicting form entries. In *Nature & Biologically Inspired Computing, 2009. NaBIC 2009. World Congress on*, pages 427–432. IEEE.

Turban, E., Aronson, J., and Liang, T. (2005). *Decision Support Systems and Intelligent Systems*. Prentice Hall, 7th edition edition.

Tuten, T. L., Urban, D. J., and Bosnjak, M. (2000). Internet Surveys and Data Quality: A Review. *Online Social Sciences*, pages 7–27.

Twitter (2016). Twitter. Available at: http://www.twitter.com.

US Census Bureau (1997). North American Industry Classification System (NAICS).

Vennjobs (2016). Startup jobs in sales, marketing, design and tech. Available online at: http://www.vennjobs.com.

Wageindicator Foundation (2016). Wageindicator foundation. Available at: www.wageindicator.org.

Wagner, C. G. (2011). Emerging careers and how to create them. 70 jobs for 2030.

Ward, D., Hahn, J., and Feist, K. (2012). Autocomplete as Research Tool: A Study on Providing Search Suggestions. *Information Technology and Libraries*, 31(4):6–19. American Library Association.

Werner, B. and Fulton, D. (2012). Best practices for web self-service user interfaces. Oracle Corporation.

William, B. F. and Baeza-Yates, R. (1992). *Information Retrieval: Data Structures and Algorithms*. Prentice Hall.

Witten, I. H. and Frank, E. (2005). *Data Mining: Practical Machine Learning Tools and Techniques*. Elsevier.

Wowczko, I. A. (2015). Skills and vacancy analysis with data mining techniques. 2(4):31–49.

Wu, X., Chen, H., Wu, G., Liu, J., Zheng, Q., He, X., Zhou, A., Zhao, Z.-Q., Wei, B., Gao, M., et al. (2015). Knowledge engineering with big data. *IEEE Intelligent Systems*, 30(5):46–55.

Yan, Y., Yin, X.-C., Zhang, B.-W., Yang, C., and Hao, H.-W. (2016). Semantic indexing with deep learning: a case study. *Big Data Analytics*, 1(1):7.

Yang, J. and Watada, J. (2011). Decomposition of term-document matrix representation for clustering analysis. In *Fuzzy Systems (FUZZ), 2011 IEEE International Conference on*, pages 976–983. IEEE.

Yang, X., Guo, D., Cao, X., and Zhou, J. (2008). Research on ontology-based text clustering. In *Third International Workshop on Semantic Media Adaptation and Personalization*, pages 14–146.

Yu, L., Hermann, K. M., Blunsom, P., and Pulman, S. (2014). Deep learning for answer sentence selection. *arXiv preprint arXiv:1412.1632*.