Ontology based Employer Demand Management
Online Employer Demand Analysis

CHAMONIX TERBLANCHE, Curtin University
PORNPIT WONGTHONGTHAM, Curtin University
BORIS MARCELO VILLAZON TERRAZAS, iSOCO

Skills shortages globally pose a real and urgent need for proper investigation and workforce development planning into the future. Analysing workforce development and employer demand needs through electronic job market allows much deeper and wider research into skill shortages. Current methods do not provide the level of depth required to address such important economic implications. In this paper we present a system aiming to gather and analyse current employer demand information from online job advertisements. It identifies current employer demand needs analysed from electronic job market.

1. INTRODUCTION

Many economies such as Australia, New Zealand, Bulgaria and Canada, have suffered severe damage to its prospects of economic growth for decades. This damage has been due to frequent periods of skill shortages (Curtain, 1988) — especially post World War I and World War II. Skill shortages occur when employers struggle to fill vacancies for an occupation, or significant specialised skill needs within that occupation, at current levels of remuneration and conditions of employment, and in reasonably accessible locations (DEEWR, 2008-09). The lack of sufficiently skilled and experienced labour has been a topic of great concern in the media for the past decade (Access Economics, 2009; Chamber of Minerals and Energy WA, 2009; MacDonald & Klinger, 2009; Probyn, 2009; Storey, 2001).

Skills shortages globally pose a real and urgent need for proper investigation and workforce development planning into the future. Analysing workforce development and employer demand needs through electronic datasets allows much deeper and wider research into skill shortages. One way to gather employer demand intelligence is by analysing online job advertisements. Adoption rates for using e-based resources to advertise and recruit future employees were already 96 per cent for North - American based global 500 companies in 2003 (JobsDB Dimension, 2011), and 75 per cent for all companies in 2008 (O'Callaghan, 2008), making online job advertisement analysis a very viable option. In general, approaches to gathering information about employment needs have been manual to date, not founded on structured frameworks, and mainly ad -hoc. Current online approaches specifically used to identify economic trends are far too high level to acquire the detailed data necessary for workforce development, skill shortages and educational need analyses. Structured electronic systems, on the other hand, do not focus on gathering data for large scale occupational and geographical workforce development needs.

This paper serves to introduce a framework to assist the identification and analysis of real-time employer demand data from a range of sources. The outcomes of the work include information of employer demand intelligence such as skill shortage indicators, workforce development information, skill set requirements for specific occupations, benefits offered by employers and the types of occupations which are in demand in specific geographical areas, etc.

In this paper we present a method and a tool named Employer Demand Intelligence Tool (EDIT) aiming to gather and analyse current employer demand information from online job advertisements. The employer demand domain

knowledge is captured in Employer Demand Ontology (EDO) and is used as the main building block of the tool. The novel aspect of this approach is the association of a domain concept with a terminology in employer demand domain. This involves finding the appropriate EDO concept for each word of a terminological string found in the online job advertisements including the appropriate conceptual relationships that hold among the concept components. Outcomes of EDIT are EDO dataset which can be used to identify current employer demand needs analysed from electronic job market.

Critical questions to be answered include:

1. What is appropriate system architecture for analysing workforce development and employer demand?
2. How can EDIT is developed as a semi-automated tool to identify current employer demand needs that are analysed from electronic job market?
3. How can the employer demand intelligence framework be evaluated for industry readiness so that it can serve as the foundation for future employer demand intelligence expansion?

This paper addresses the first question above by providing a framework for employer demand intelligence that addresses the research goal to develop a semi-automated tool to gather, analyse, and report on employer demand intelligence. This is presented in sections 3 and 4 in this paper. The second question above is addressed by developing EDIT which is presented in section 5 in this paper. The third question above is addressed in this paper by verifying the framework in a case study and validating the EDIT through a set of queries. This is presented in section 6 in this paper.

The overarching objective of this research is to develop a framework for employer demand intelligence which ensures that ample breadth and depth of employer demand intelligence concepts are captured. This vast framework will require a semi-automated system to ensure continuous population of up-to-date employer demand intelligence. In order to proceed with such task, a systematic and logical approach needs to be followed. This will ensure that the framework's development is based on a quality scientific method.

The remainder of the paper is organised as follows. Section 2 presents literature review. Section 3 introduces the EDO. Section 4 gives an overview of EDIT architecture. Section 5 outlines EDIT prototype development for proof of concept. Section 6 provides verification of the framework and validation of EDIT. Section 7 summarises limitations of EDIT and also sheds light on possible future work. Section 8 concludes the paper.

2. RELATED WORK

There are various approaches being used to identify employer demand needs. Current providers of employer demand data in Australia include the Australian Bureau of Statistics[1], the Department of Employment, Education and Training, employer associations, unions, state and territory employment and training bodies, industry training committees and TAFE (Hayton, 1988). No one study's outcomes should be used as the only data source for determining which skills are in shortage at present. There is a great need for studies that gather much wider, deeper and more accurate data about the needs that employers have for specific occupation types and skill sets in their organisations. Many stakeholders have mentioned the need to accurately gather and analyse employer demand data in order to obtain a better

---

[1] http://www.abs.gov.au/

picture of up-to-date employment needs. Such intelligence would greatly assist workforce development, immigration, education and training policy development, and individual career and organisational human resource planning in the future.

Current approaches mainly focus on gathering employer demand data through conducting surveys with employers, or by looking at high-level online data about the number of job advertisements from one period compared with another. Governments mainly utilise data obtained through surveys and high level job advertisement counts. Other approaches consist mainly of job analyses done by academics to obtain one-off intelligence about a specific occupation. Existing job advertisement analyses do not cater for detailed workforce development, educational and immigration decision needs.

The uptake of eRecruitment tools and research foci has been varied and can be broadly classified. The most common purposes of existing semantic approaches are either to classify an organisation's internal human resource needs, to do job matching between the organisation's vacancies and job seekers' resumes, or to do competency management. Table 1 lists and summarises the existing semantic approaches.

Table 1: A Summary of all structured employer demand identification approaches analysed as part of the EDIT project.

Source	Ontology purpose	Existing standards used?	Country	Occupation/ Industry focus
(Lau & Sure, 2002)	Competency management	No	Switzerland	Private insurance, human resources, IT
(Trichet & Leclère, 2003)	Job match and competency management	No	France	IT
(Biesalski & Abecker, 2005)	Competency management	KOWIEN	Germany	Automotive industry
(Gualtieri & Ruffolo, 2005)	Generic Organisational HR representation	No	Italy	NA
(Terziev, Kiryakov, & Manov, 2005)	Generic Organisational HR representation	No	European	NA
(Bizer et al., 2005)	Job matching	HR-XML, KOWIEN American & German occupational classification systems, industry classification systems.	Germany	All
(García-Sánchez, Martínez-Béjar, Contreras, Fernández-	Job matching	No	Spain	Rural sector occupations

Breis, & Castellanos-Nieves, 2006)				
(Lv & Zhu, 2006)	Job matching	No	China	Unclear but seems to be related to ICT.
(Dorn, Naz, & Pichlmair, 2007)	Job matching and competency management	HR-XML	Austria	University student skills
(Gómez-Pérez, Ramírez, & Villazón-Terrazas, 2007)	Job matching	KOWIEN and 11 other standard classification systems	Europe	ICT
(Zemmouchi-Ghomari, 2012)	Generic Organisational HR representation	No information available to determine this.	Algeria	University staff

Human Resource Representations: The PROTON knowledge management structure (Terziev et al., 2005) is a multi-layered ontology with a general purpose, domain-independent ontology intended to be reused for the sake of consistency. Another human resource representation has been built to model the human resource structure of a Higher Education organisation in Algeria (Zemmouchi-Ghomari, 2012), and on describing the role of each university employee and student in relation to the university as a whole.

Job Matching Approaches: Several studies have focused on providing a job-matching service, mostly for industries related to Information Technology. Organisations generally post their vacancies on a web-server somewhere, and job applicants then post their résumés to the call-for-applications portal or email address listed on the job advertisement. The job-matching ontology systems are intended to assist in classifying each vacancy and résumé into specific ontologies, and subsequently performing a matching exercise to relieve the major task of manually screening the applicants. The biggest job-matching project to date has been the Single European Employment Market Place (SEEMP) project in Europe (Gómez-Pérez et al., 2007). The project's aim was to deliver a system that was based on platforms and networks between different online recruitment websites for the information communication and technology sector. Similarly, the Knowledge Nets system (Bizer et al., 2005), developed for the German market, was developed to assist employers to post job offers more efficiently. The EXPERT system (Kumaran & Sankar, 2013), on the other hand, was developed to map job requirements to candidate skills and highlight eligible candidates.

Competency Management Approaches: Competency management approaches generally aim to compare an organisation's existing human resources pool to the preferred or required human resource pool, in order to ensure that the company functions at its greatest efficiency. These studies are also referred to as gap analyses. One such system was developed for Swiss Life (Trichet & Leclère, 2003), a large insurance company based in Switzerland, and another was established in France for

the Information and Telecommunication Technology (ITT) domain (Trichet & Leclère, 2003).

Current approaches have been designed for other purposes. The eRecruitment structures, to date, have been designed for three distinct purposes: i) to provide human resources templates for the development of organisational structures; ii) to match job offers with job seekers; and iii) to compare an organisation's current human resource set of competencies with its ideal set of competencies. Whereas point i) above does not aim to gather any data around its structure, but is merely provided to assist the development of future structures related to human resources, points ii) and iii) involve temporarily gathering data for the purposes of an ad hoc matching or comparison. Structures designed for the purposes of job matching, for example, would record the occupational requirement that the job seeker has provided and match it with any job offers that have the same occupational vacancy type as that which the job seeker has indicated; the structure allows the matching to take place, and thus the process has been completed. Structures built for the purposes of competency management (point iii) above) record the skill set information about an organisation's current human resources pool and then compares it with the organisation's ideal pool of human resources skill sets. This is another matching process to find the same type of information, generally involving either one individual or one company only. A detailed employer demand intelligence structure, on the other hand, requires a structural design that can gather, continuously analyse and permanently keep various types of information at different points in time, across geographical, industrial, occupational and sectoral borders.

None of the existing approaches listed above has been maintained or is still operational. Some of the approaches merely made it to a 'pen on paper' stage; others were utilised by industry at some point in time, but only for a short period and have not been used again. Technological and other scientific advances have resulted in labour market changes happening at a much more frequent pace these days. Quicker travelling times, electronic communication, and machinery advances have drastically changed the ways we perform our jobs and continue to modify what we do. In order to reflect these significant changes and accurately identify current trends, it is imperative that human resource structures are kept up-to-date and in line with these changes.

Current structured approaches do not include the ability to record the period of time that an advertisement was published for. The period a vacancy remains vacant is a significant indicator of a possible skill shortage (Green, Machin, & Wilkinson, 1998) and hence imperative to be recorded alongside other vital skills shortage and employer demand indicators.

3. EMPLOYER DEMAND ONTOLOGY

Among the computer science and artificial intelligence circles, an ontology is most commonly known as 'a formal and explicit specification of a conceptualisation' (Gruber 1993), which allows information owners to annotate their material (Berners-Lee, Hendler et al. 2001). Several definitions of ontology have been provided to date (Neches, Fikes et al. 1991; Studer, Benjamins et al. 1998; Maedche 2003) however the core of them all relate to the usefulness of ontology in technology for the representation and sharing of knowledge about a domain, by modelling concepts and the relationships among those concepts within the domain (Gruber n.d.).

With the acceptance that an ontology is based on the notion of a shared conceptualisation, users, designers and domain experts need to agree on the specific knowledge contained in any ontology to deem the ontology usable. To assist in this task, different layers of ontology have been introduced to eliminate disagreement and

foster the re-use of ontologies as opposed to reinventing new ontologies each time. The three layers of ontology knowledge, based on their generality, are (Chang, Dillon et al. 2006; Guarino, Oberle et al. 2009):

- Top level ontologies, which relates to general concepts that are domain independent (e.g. space, time etc.)
- Domain ontologies, which capture the general knowledge of a specific domain or task. In this research, it is the employer demand domain where general concepts such as competence and geographic concepts are captured.
- Application ontologies, which represent concepts that capture the knowledge necessary for a specific task or application within a domain. In this research, it is the occupation employer demand domain relating to specific job offers that are included as concepts of this layer.

Employer Demand Ontology (EDO) represents a shared understanding of the employer demand domain knowledge enabling efficient and intelligent management of all its existing information sources across the domain. EDO has been developed in Web Ontology Language (OWL). As recommended by the World Wide Web Consortium, OWL is considered as the standard semantic markup language for publishing and sharing ontologies. Using this standard language for developing EDO is necessary for interoperability with existing ontologies in future work.

Ontology notation is a matter of taste and has been the cause of many arguments (Sowa, 2009). As a result, many notations exist with none being 'right' or 'wrong'. The EDO uses a variation of the Software Engineering Ontology developed by Wongthongtham (Wongthongtham, 2006). For classes, a double field box is used where the top compartment hosts the mandatory term <<Concept>>, accompanied by the actual class label or concept's name, e.g. *JobAdvertisement*. The bottom compartment is provided to indicate the properties belonging to that class. An instance, on the other hand, is represented as an oval shape containing the instance label or name for both class and property instances. The oval is attached to either a class or property via an arrow; a closed arrow head indicates a specialisation/ generalisation relationship where an open arrow head indicates a composition relationship (see Figure 1).

| <<Concept>> X | Y | Specification / Generalisation | Composition |
| Ontology Class X | Ontology Instance Y | Relationship | Relationship |

Fig. 1.Ontology Notations

The EDO naming convention is similar to that used by Terziev, Kiryakov et al. (2005). Class names are capitalised, and when comprising more than one word, each subsequent new word is also capitalised without any spaces or alphanumeric symbols in between (e.g. *PrintedMediaAdvertisement*). Relation and attribute labels follow the same principle as that for class labels except for the first letter of the label that is not capitalised (e.g. *hasAdvertisementStartDate*).

EDO consists of two tiers as shown in Figure 2. Tier one is a high level, non-specific occupation type template for general concepts that are applicable to all occupation types. Tier two is a detailed occupation-specific subsection template relevant to each separate occupation as defined under the ANZSCO.
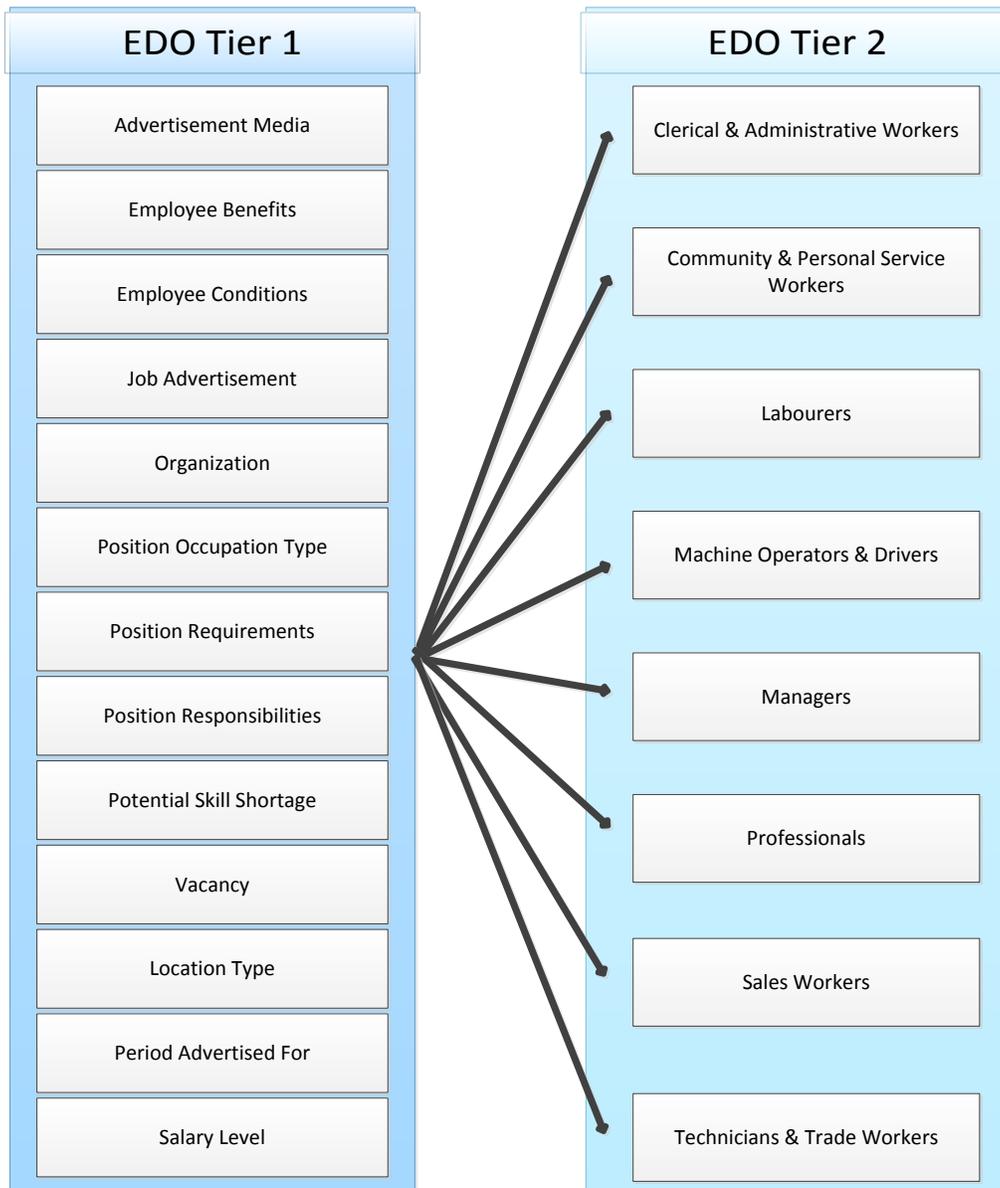
Fig. 2. Employer demand ontology tiered framework

Vacancy: A job opening that is offered by an employer.

Job Advertisement: A vacancy that has been advertised via some type of advertisement media. Direct information captured in a job advertisement usually includes what the position requires from a prospective employee, the responsibilities that that position entails, the salary on offer to the successful candidate, the organisation where the vacancy exists, the job title of the future employee, where the position is physically located, the benefits on offer to the future employee, and the conditions under which the person can be expected to work. Indirectly, a job advertisement could signal possible skills shortages that exist for the specific occupation type in an exact geographical area (see Potential Skill Shortage class below). A job advertisement may also indirectly provide information about the length of time that the vacancy was advertised (see Period Advertised class below). Figure 3 shows the composition ontology for a job advertisement in EDO. The colour coding of

concepts in the figure denotes that they have subclasses; each subclass composition has been further detailed in the relevant sections below coded in the same colour.
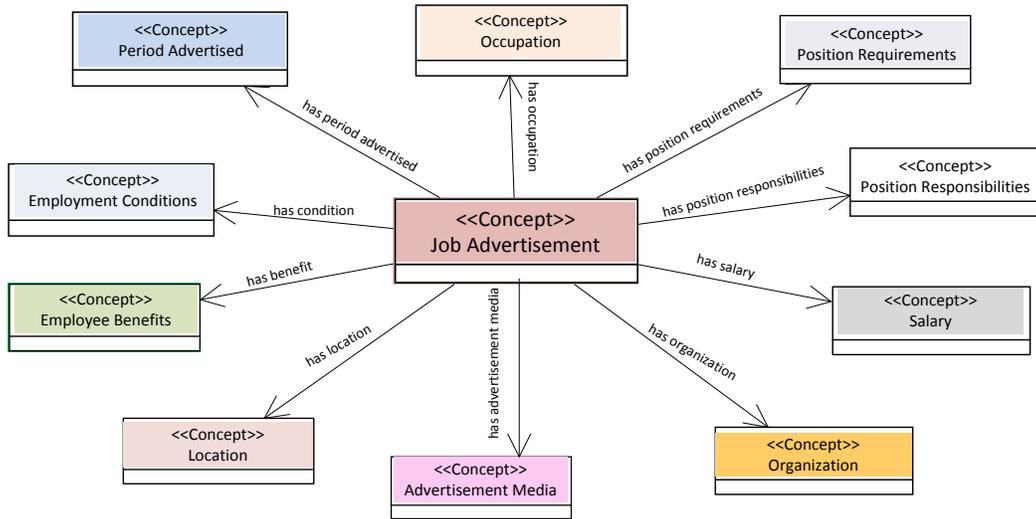
Fig. 3. Job advertisement composition ontology

In Figure 4, the example of a job advertisement instance is reflected for the job advertisement MW11 which is the unique number assigned to a job advertisement relating to a midwifery position, and the eleventh advertisement of this kind that was captured and recorded in the employer demand knowledge base.
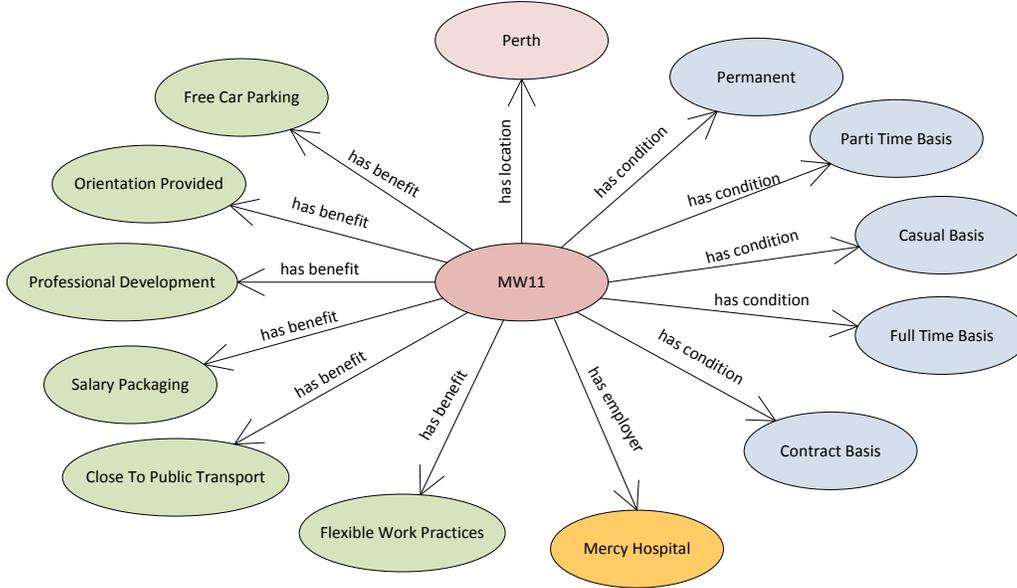
Fig. 4. Job advertisement instance ontology example.

Advertisement Media: The media through which the employer decides to advertise the vacancy that currently exists at their company or institution. Examples include company job websites, LinkedIn advertisements and various types of newspaper advertisements (Figure 5).
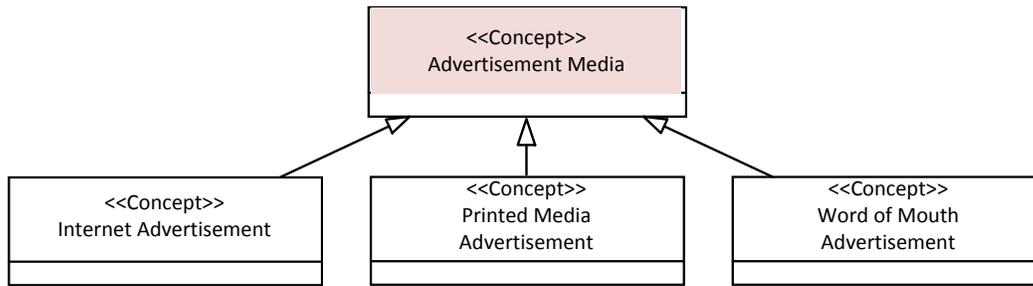
Fig. 5. Advertisement media generalisation-specification ontological hierarchy.

Period Advertised: This class structure indicates the period of time for which a position has been advertised and can be subdivided into three subclasses: short, average and long. Counting the days from the start to the end date that the job was effectively being advertised provides the period (number of days/ weeks) during which potential employees were invited to apply for the position. In the EDO, a vacancy has been advertised for a short period if it was advertised for only one week or less. When an advertisement runs between 1-4 weeks, it is considered an average period, and when it runs for longer than 4 weeks, it is considered a long period. Figure 6 provides a hierarchical view of this class.
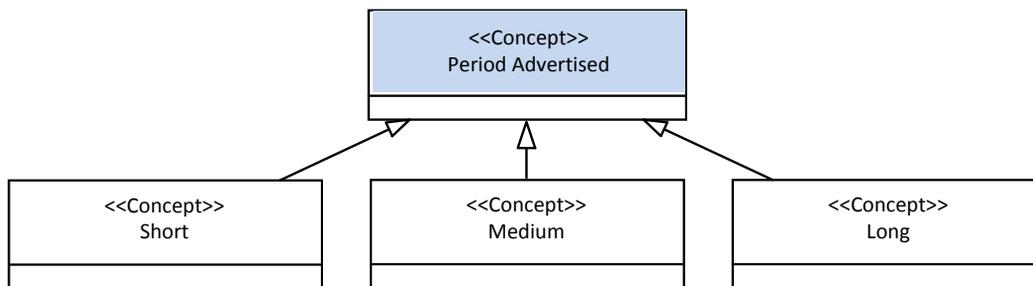
Fig. 6. Period advertised hierarchy.

Position Requirements: The abilities that an employer expects the employee to have (Figure 7). There are three subclasses for position requirements (attribute, competence and other requirements) and a further five subclasses for competence requirements (experience, knowledge, language, skills, qualifications, licence(s) or registration. Examples of attribute requirements include enthusiasm, passion and reliability. Examples of other requirements include physical fitness, having a police clearance and working rights. Examples of competence requirements include, for example, having experience working with budgets, knowledge about paediatrics, being able to speak French, having a tertiary qualification, and planning skills.
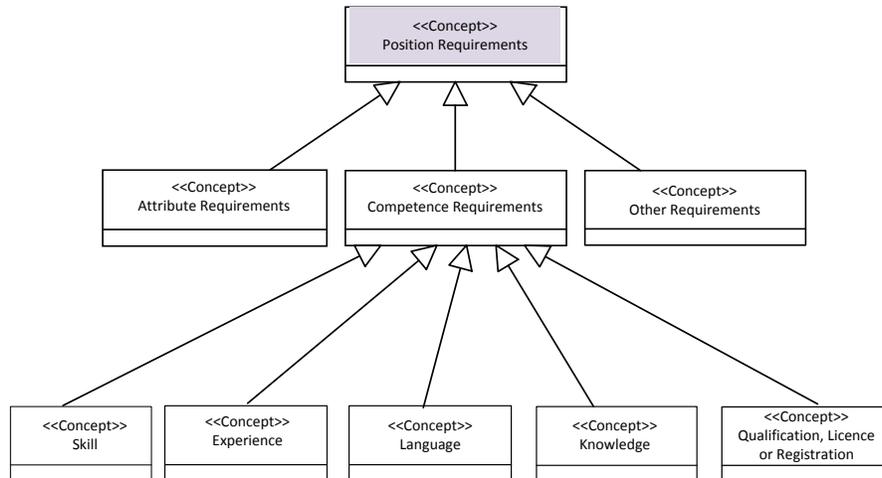
Fig. 7. Position Requirements ontological hierarchy.

Position Responsibilities: The tasks that the employee will be expected to perform, e.g. organising day trips, doing filing or managing a team.

Salary: The remuneration that will be paid to the employee.

Organisation: The organisation that has the vacancy.

Occupation Type: The type of occupation for which the vacancy exists. The list of occupations has been adapted from the ANZSCO which is specific to the Australian market. Further information is provided in the 'evaluation' section below.

Location Type: The location type indicates the Remoteness Area of a geographical location as determined by the Australian Bureau of Statistics. The Australian Standard Geographical Classification (ASGC) Remoteness classification was developed by the Australian Bureau of Statistics (ABS) in response to a demand for a statistical geography that allows quantitative comparisons between 'city' and 'country' Australia where the defining difference between 'city' and 'country' is physical remoteness from goods and services. The remoteness of a location is very important in Australia due to typically very long distances that need to be travelled from major cities to other areas to provide essential services for these remote locations. The ABS classifies remoteness in the Accessibility/Remoteness Index of Australia (ARIA) which comprises of six Remoteness Areas (RAs): i) Major cities of Australia, ii) Inner Regional Australia, iii) Outer Regional Australia, Iv) Remote Australia, v) Very Remote Australia, and vi) Migratory. As the last category relates to countries other than Australia, this category was not included in the development of the Location Type class in the EDO.

Employee Benefits: Benefits that the employee will receive while employed in this position. Examples include having a company car to drive with to work, receiving free flights and being granted qualification allowances.

Employment Conditions: Conditions relating to the vacancy that indicate the work hours and position basis of the vacancy. Position bases options include being employed on a casual basis, a full-time basis or a relieving basis. The concept of work hours covers things like having to work weekend shifts, being required to travel or being able to work only relief shifts.

Potential Skill Shortage: 'Skill shortages exist when employers are unable to fill or have difficulty in filling vacancies in a recognized occupation for specialisation at the current level of remuneration and conditions of employment, including realistic location' (Department of Employment Education Training and Youth Affairs

1998). The EDO's potential Skill Shortage Vacancy concept has been designed to capture vacancies that include the skill shortage indicators mentioned in this definition. Thus, when a vacancy is identified as meeting two of the mentioned three conditions in the definition, the EDO will flag it as a potential skill shortage for investigation. Figure 8 provides a graphical illustration of the EDO's potential skill shortage identification basis. The EDO also assumes that 'realistic locations' relate to metropolitan regions, and that the other four regional classifications found in the ASGC (inner regional, outer regional, remote and very remote) could all indicate a struggle to find the relevant employees for that specific area if vacancies offer a high remuneration and are advertised for a long period of time.

Based on the EDO's potential skill shortage indicators, some of the scenarios that could be captured by the EDO as a potential skill shortage are listed below. Clearly, many more scenarios are possible.

- Remote location, high salary level, long period advertised
- Outer Regional locations, high salary level, average period advertised
- Inner Regional location, average salary level, long period advertised
- Very remote location, average salary level, average period advertised
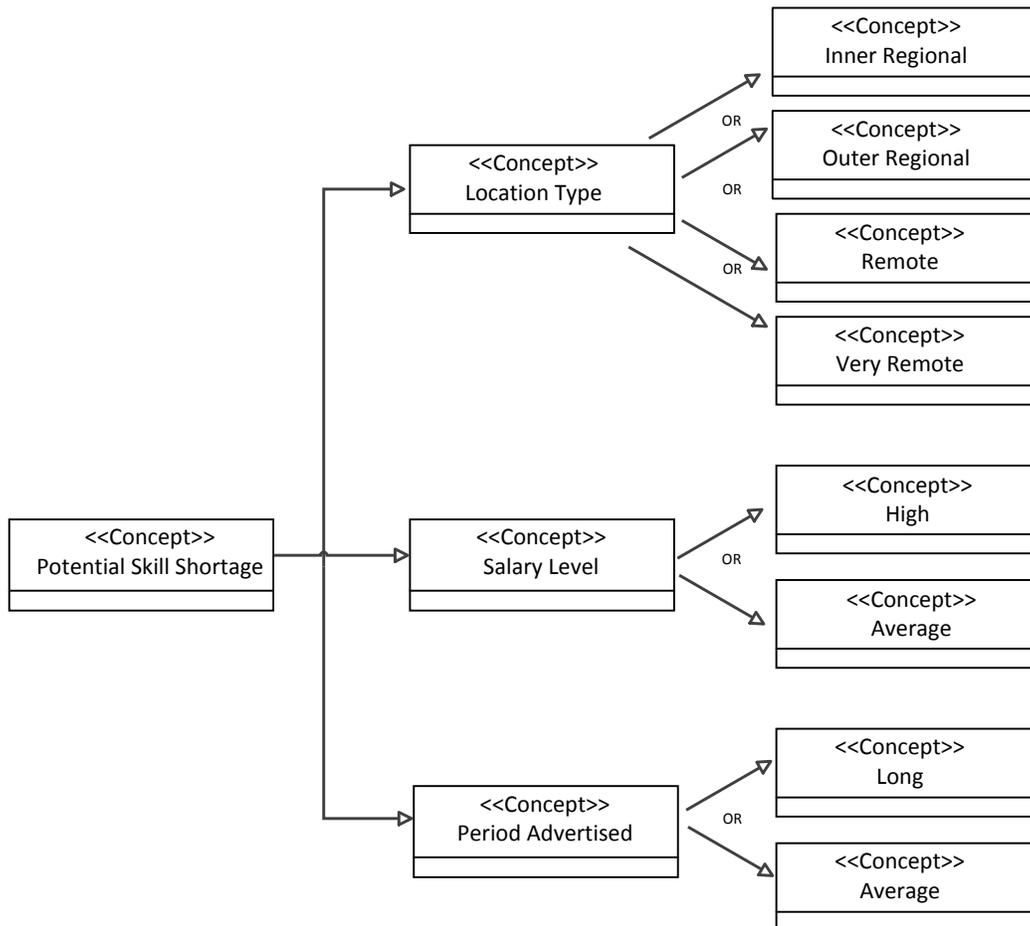
Fig. 8. Potential skill shortage hierarchy.

The instance population will happen through collecting one month's job advertisements from the job board SEEK2. The specific occupations, that the data will be collected for, will be for the sixteen Nursing and Midwifery occupations from the ANZSCO. The geographical area that will be covered will be that of Western Australia, Australia's largest geographical state, where Perth is the capital city. An extension of this geographical parameter is intended as future work to also include other Australian states and eventually other countries too.

4. ARCHITECTURE OF THE EMPLOYER DEMAND INTELLIGENCE TOOL (EDIT)

EDIT aims to overcome the issue whereby current employer demand related approaches are extremely resource intensive, causing huge time delays in gathering relevant and current data related to the domain. EDIT enables to semi-automatically populate instance knowledge extracting from online-published job advertisements. It first extracts job advertisement content from online sources filtering unwanted web content. It then finds the appropriate EDO concept for each word of a terminological string found in the online job advertisements. Next it links the recognised word to the corresponding EDO concepts. It also asserts relationships among the concepts. In the prototype for proof of concept, inputs are job advertisements collected for Western Australian locations and for the Nursing and Midwifery occupations; outputs are EDO dataset which can be used to identify current employer demand needs analysed in real time from online sources. It is very important to have a clear picture of the job market at any time — not only for governments for timely decision making but also for scenarios like individuals doing career planning. Furthermore, due to the volatile and dynamic nature of the job market it is necessary that the information about employer demand should be up-to-date. This information can be collected from online job advertisements. However, the job advertisements usually contain unstructured information in textual form with frequent usage of specific jargon related to the nature and field of the advertised vacancy. Additionally, the information collection involves hundreds of advertisements, which is difficult to process manually, if possible at all. The advertisements, albeit highly unstructured, are already in digital form and the development of an automated process capable of extracting meaningful structured data and statistics for these text advertisements can be extremely useful in this scenario.
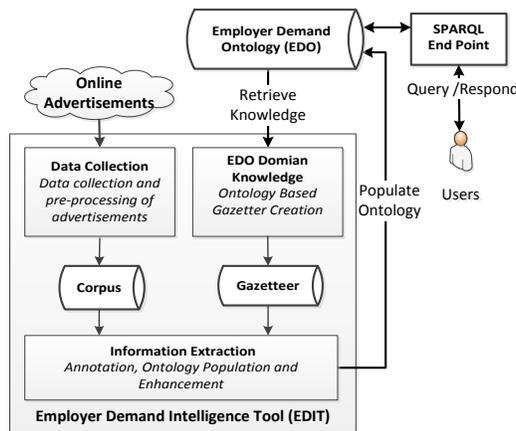

Fig. 2. Architecture of Employer Demand Intelligence Tool (EDIT)

---

2 www.seek.com.au

Figure 2 shows the architecture of the EDIT. It starts where online job advertisement content is collected and stored in a corpus. Unnecessary content such as links and graphics is removed. The extraction of information contained in the corpus is done next. In addition to the corpus, gazetteer is created as domain knowledge source from the EDO. It extracts the EDO concepts (EDO classes and instances) and stored them in the gazetteer. The information extracted from each advertisement is then populated into the ontology and appropriate conceptual relationships among concept components are also populated. As such, the EDO is enhanced and evolved although no concepts and relationships are removed from EDO. The extended EDO forms as knowledge base in which end users can query through SPARQL endpoints in order to obtain employer demand intelligence interrogations.

5. DEVELOPMENT OF EDIT

This section focuses on the EDIT prototype development. EDIT has been developed using GATE (General Architecture for Text Engineering) API and plugins for all the required functionality for text and ontology processing. The reason for using GATE as opposed to OINTED (Wibisono et al., 2013),OWL2RDB (Stanimirović, Bogdanović, & Stoimenov, 2013), or any other tool to populate ontology from unstructured to structured data, is twofold. Firstly, GATE is an open source software and has been used for many research-oriented projects (Cunningham et al., 2013). Secondly, GATE provides a development environment which is needed in this project to develop a natural language processing module. The input of a process is a language resource. The process can operates one or more language resources to produce the desired output. In EDIT, we have two language resources which are processed. The first language resource is the individual document in corpus of documents. It contains all the online job advertisements collected. The second is EDO based language resource. Knowledge contained in the EDO is used to extract information from the job advertisements in the corpus and populate this extracted information back to the EDO which then can be accessed using standard SPARQL queries. Figure 3 shows the EDO loaded into EDIT as a language resource. EDO plays a dual role in EDIT: firstly it is used to generate a knowledge base and at the end of the process it is once again used by EDIT to store the information that is automatically extracted from the documents (job advertisements).

Fig. 3. The EDO loaded as a language resource into the EDIT (GATE application)

Figure 4 shows the sequence and flow of processes in EDIT.
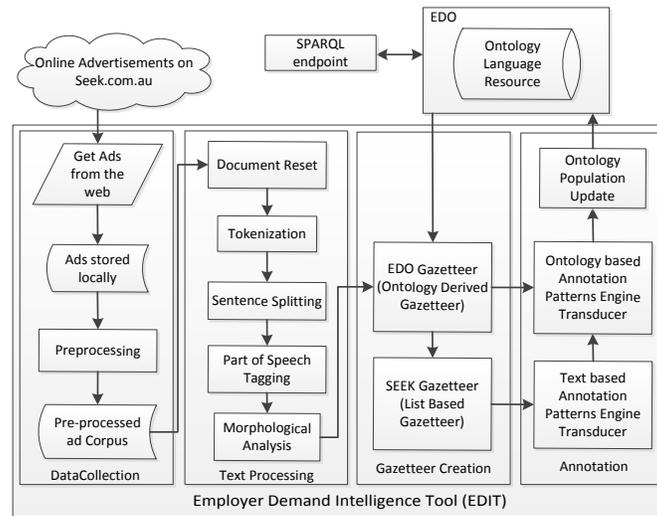
Fig. 4. Employer Demand Intelligence Tool (EDIT) flowchart

5.1 Data Collection

Archiving Job Advertisement

Job advertisements that are published on the online job portal SEEK were downloaded and stored on a daily basis. The process was repeated daily and the advertisements were organised into corpora of advertisements published online in a calendar month. Using the standard terminology in text processing, the term 'document' is used to refer to each job advertisement. The documents are saved in html archive files with all the webpage graphics and links. This raw data is kept for future referencing purposes.

Pre-processing Job Advertisement

Web Images and links which were not directly related to the advertisement were removed by our system using the Document Object Model (DOM). It parsed to remove these images and links while retaining the job advertisements' text.

5.2 Text Processing

Document Reset

Document reset process resets a document to its original state by removing all the annotation sets except the one containing the document format analysis (as original mark-ups). This is to make sure that it will not run on documents which have already been processed. We use A Nearly New Information Extraction (ANNIE) Engine for document reset process which is a part of the GATE component.

Tokenizer

Tokenizer process breaks up the text into tokens. Meaningful elements within text such as words, phrases or symbols are called tokens. The task of identifying the tokens and breaking up a stream of text into tokens is called tokenization (Wilcock, 2009). A tokenizer can split text into tokens by searching for white space (spaces, tabs and newlines) in the text. However, the presence of punctuations, numbers and symbols etc. in the text needs to be accounted for and complex string processing with

regular expressions is needed for meaningful tokenization. We adopt and modified GATE tokenizer called the ANNIE English Tokenizer.

Sentence Splitting

In this process, the sentence boundaries are determined. Sentence boundary detection, which entails finding the point in the text where the sentence starts and the point where it ends, is performed. The simplest approach for written language is to split the text at full stops ("."). However, this is too simple, as there are other end-of-sentence markers besides fullstops, such as question marks ("?") and exclamation marks ("!"). Moreover, other uses of full stops that do not mark the end of a sentence, such as in abbreviations ("Mr.", "etc."), also exist. Another scenario is where a full stop has more than one function, for example when "etc." is the last word of a sentence — a single full stop can simultaneously mark both the end of the abbreviation and the end of the sentence. A practical reason for exercising sentence boundary detection is that syntactic parsing is normally done on individual sentences rather than complete texts. The time that automatic syntactic parsing takes as the text length grows, seems to increase considerably in a computational context. Therefore, it is important to split the text into shorter units at the onset (Wilcock, 2009). We employ in EDIT the ANNIE Sentence Splitter to segment the text.

Part Of Speech (POS) Tagging

In POS tagging process, it determines the usage of a word as noun, verb, adjective, adverb, pre-position, article or determiner, conjunction, and interjection. Linguistically these categories have been further subdivided; subdivision details can be found at x. The POS tagger assigns each token with a part-of speech label. We utilise the ANNIE POS Tagger in EDIT by using a default lexicon and a rule set (x).

Morphological Analysis

In morphological analysis, the sentence structure is analysed for identification, analysis and description of the structure of a given language's linguistic units, such as root words and affixes. We use Gate Morphological Analyser in EDIT.

5.3 Gazetteers Creation

Gazetteer is a list that is used to find named entities in text. For example a list of names of places in Australia can be used to find the mention of a particular geographical location, such as Perth in the text. We use two gazetteers in EDIT, the first is EDO gazetteer which is a created flexible gazetteer from the knowledge contained in EDO by using the GATE OntoRoot component and the second is SEEK gazetteer which is a created word-list based gazetteer.

EDO Gazetteer (Ontology Derived Gazetteer)

The Flexible Gazetteer is used in EDIT to employ a list of entities generated from an ontology instead of a using a list of names (like the ANNIE gazetteer). The Flexible Gazetteer utilises the list of names produced at runtime by another EDIT component called the OntoRoot Gazetteer. The OntoRoot Gazetteer takes an ontology as the input and constructs a list of names for the Flexible Gazetteer's use. In EDIT, EDO is provided as the input to the OntoRoot Gazetteer and links with the Flexible Gazetteer. The Flexible Gazetteer then looks in the documents (job advertisements) for the words and phrases provided by the OntoRoot Gazetteer. The output of this contains the words or phrases which appear in the ontology as instances of classes and the location of the matched text within each document.

SEEK Gazetteer (List Based Gazetteer)

The ANNIE Gazetteer is used and modified (coined the SEEK Gazetteer) for use in EDIT. This is a truncated version of the original ANNIE Gazetteer which is used to recognise other named entities that are not mentioned in EDO.

5.4 Annotation

In text processing, transducers are finite state machines that process annotations by using regular expressions. We utilise a Java Annotation Patterns Engine (JAPE) to implement these transducers. The JAPE Transducer operates over annotations based on regular expressions. It is used to perform certain actions on the output of previous processing resources. Each JAPE grammar has two parts, which are separated by a '->' symbol. The first part is a regular expression and the second part is a Java program which performs certain actions. The first part (the regular expression) processes the input (tokens); and when a token or sequence of tokens match with the regular expression, the Java code in the second part is executed to perform the desired processing. EDIT employs two JAPE transducers for JAPE grammar rules to extract information from the documents, based on the SEEK Gazetteer, while the other is an ontology aware JAPE transducer.

The grammar shown in Figure 5 is part of the JAPE transducer's processing of the output as part of the non-ontology aware gazetteer. This process searches for the concept of 'salary' in each job advertisement.

```
Phase: Salary
Input: Token Money
options: control = first


Rule: salaryDual
({Token.string == "Salary"} {Token.string==":"} {Money}{Token.string == "-"} {Money})

:lookup
-->
:lookup.Benefit = {kind="Benefit", rule="salaryDual"}


Rule: salarySingle
({Token.string == "Salary"} {Token.string==":"} {Money})
:lookup
-->
:lookup.Benefit = {kind="Benefit", rule="salarySingle"}
```

Fig. 5. Program code to extract salary amounts from job advertisement text

Another grammar is show in Figure 6, where the code looks for the dates that are contained in the job advertisement. This grammar populates the ontology with the advertisement dates of the job advertisement, and is a part of the ontology-aware transducer.

```
// This grammar checks whether or not the document has an instance in the ontology
// and then puts the advertisement date as a data type property value.

Imports: {
import java.net.URL;
import org.apache.commons.io.FilenameUtils;
}
Phase: InsertDate
Input: Date
Rule: insertDate
({Date}):mention
-->
:mention{

String baseURI="http://www.semanticweb.org/ontologies/2011/7/EmployerDemandOntology.owl#";
URL docURL=doc.getSourceUrl();
String baseName=FilenameUtils.getBaseName(docURL.getFile())+"."+FilenameUtils.getExtension(docURL.getFile());


// initializaiton
gate.AnnotationSet lookup = (gate.AnnotationSet) bindings.get("mention");
gate.Annotation mentionAnn = (gate.Annotation) lookup.iterator().next();
FeatureMap lookupFeatures = mentionAnn.getFeatures();
gate.FeatureMap features = Factory.newFeatureMap();
features.putAll(lookupFeatures);

//find the text covered by the annotation
String theMentionText = gate.Utils.stringFor(doc, mentionAnn);

 // verifying the existence of class "Advertisement" in the Ontology
 OClass aClass = ontology.getOClass(ontology.createOURI(baseURI+"Vacancy"));

 //create URI for current document
 OURI jobURI = ontology.createOURI(baseURI + baseName);

 //search the instance in ontology
 OInstance jobInstance=ontology.getOInstance(jobURI);
 //System.out.println("Instance Found:"+ jobInstance);
```

Fig. 6. Program code to detect advertisement date and populates the ontology with the advertisement dates of the job advertisement

5.5 Ontology Population Update

The ontology aware JAPE transducer is used to populate EDO with the information extracted. The transducer then verifies the information extracted by the Flexible Gazetteer and creates the necessary instances, object property values and datatype property values in EDO. In Figure 7, the advertisement text which matches the concepts contained in EDO, is highlighted. For example, the concepts *Bunbury* and *Harvey* were identified in the job advertisement shown in the screenshot, as instances of the greater concept *LocationType*, and the requirements of *PoliceClearance* and *ReliableTransport* were identified as they are instances of the greater concept or class called *PositionRequirements*. This matching is done by the Flexible Gazetteer.

21 Mar 2013 Location:
Bunbury & South West

Salary:
$20 - $24.99 per hour

Work type:
Part Time

Classification:
Healthcare & Medical Nursing - Aged Care

Aged Carers
$23+ Per Hour plus Allowances
Facilities in Bunbury, Eaton and Harvey
Regular shifts available
Start ASAP!!

The Company:

Flexi Staff Pty Ltd is a leading Western Australian owned and managed company specialising in all aspects of the recruitment and labour hire.

We are a major supplier in providing staff to all Aged Care facilities in Bunbury and the South West region and are currently seeking experienced Aged Care workers for regular, flexible and casual work.

We have shifts available in the Morning, Afternoons and Nightshifts and these are available Monday - Friday and Weekends.

Essential Criteria:

Cert III Aged Care (Essential)
Current Police Clearance or willing to obtain
Current WA C Class Drivers Licence
Reliable Transport
Available to start ASAP

If you are interested in this position and meet the requirements click 'APPLY' to send us your up to date resume, or for more information please call Flexi Staff Bunbury on 9791 5032
Flexi Staff Pty Ltd is an Equal Employment Opportunity (EEO) employer. Flexi Staff is committed to improving employment outcomes for all our candidates. We look to assist candidates find employment opportunities on merit.

Flexi Staff embraces the principle of diversity, and seeks to recognize and benefit from the new ideas, different ways of working and decision-making that are afforded by a diverse workforce.

Fig. 7. Text appearing in job advertisements matched with the concept in the EDO by the EDIT

In Figure 8, the JAPE grammar is shown, which creates an instance for each advertisement. This grammar checks whether an instance already exists in EDO for a job advertisement, and in case the job advertisement is new and has not been processed before by EDIT, it creates a new instance for it in the ontology. In Figure 9, a screenshot of EDO is shown, in which the instances created several job advertisements that are visible. More specifically, it also shows the object properties and data type properties (dotted line box) that were created for the advertisement '9.htm'.

```
// checks whether or not the document has been processed before
// and creates an instance of class "Vacancy" for the for the current document if it does not exist
// in the ontology.

Imports: {
import java.net.URL;
import org.apache.commons.io.FilenameUtils;
}
Phase: CreateInstance
Input: Lookup
Rule: CreateInstance
({Lookup.type == instance}):mention
//({Lookup}):mention
-->
:mention{

//System.out.println("Instance?:"+doc.getFeatures());

if (doc.getFeatures().containsKey("InstanceCreated"))
{
//System.out.println("Instance exists");
}
else
{
System.out.println("Instance does not exist. Creating ...");
doc.getFeatures().put("InstanceCreated", "yes");
String baseURI="http://www.semanticweb.org/ontologies/2011/7/EmployerDemandOntology.owl#";

URL docURL=doc.getSourceUrl();
String baseName=FilenameUtils.getBaseName(docURL.getFile())+"."+FilenameUtils.getExtension(docURL.getFile());
//System.out.println("=========================== \n Document:"+ baseName  +"\n");

// verifying the existence of class "Advertisement" in the Ontology
OClass aClass = ontology.getOClass(ontology.createOURI(baseURI+"Vacancy"));
//System.out.println("ClassFound:"+ aClass +"\n");

//create uri for current document
OURI jobURI = ontology.createOURI(baseURI + baseName);

//search the instance in ontology
OInstance jobInstance=ontology.getOInstance(jobURI);
//System.out.println("Instance Found:"+ jobInstance +"\n");

//if instance does not exist create one
if (jobInstance == null)
  //if (!ontology.containsOInstance(jobInstance))
{
//System.out.println("Creating Instance:" +jobURI+ "\n");
jobInstance = ontology.addOInstance(jobURI, aClass);
//System.out.println("Instance Created:" + ontology.getOInstance(jobURI));
 }
```

Fig. 8. Java code to create instances for job advertisements
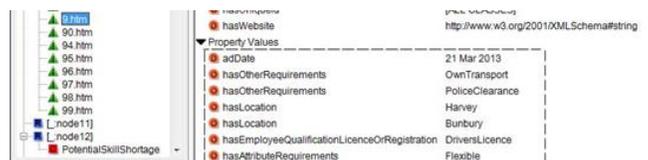
Fig. 9. New instances and properties added into EDO by EDIT (view from GATE)

6. VERIFICATION AND VALIDATION

We verify the framework in a case study and validate the EDIT through a set of queries.

6.1 Case Study

When the system recognises the occurrence of EDO captured concepts in the job advertisements, by matching them, it creates an instance for each advertisement in the EDO. Then, the additional concepts recognised are linked to the instance of the advertisement through the relationships i.e. object properties and data type properties. The system has the ability to create new data type or object properties in the ontology for those instances for which there is no matching or appropriate relationships. As such, an enhanced ontology evolves alongside the ontology population process. A case study is provided to illustrate the processes that a single job advertisement undergoes during the different stages of the framework.

Data Collection Process

Figure 10 shows an example of a simplified job advertisement after graphics and links removed in the pre-processing stage.

21 Mar 2013
- Location:
  Bunbury & South West
- Salary:
  $20 - $24.99 per hour
- Work type:
  Part Time
- Classification:
  Healthcare & Medical Nursing - Aged Care

Aged Carers

- $23+ Per Hour plus Allowances
- Facilities in Bunbury, Eaton and Harvey
- Regular shifts available
- Start ASAP!!

The Company:

Flexi Staff Pty Ltd is a leading Western Australian owned and managed company specialising in all aspects of the recruitment and labour hire.

We are a major supplier in providing staff to all Aged Care facilities in Bunbury and the South West region and are currently seeking experienced Aged Care workers for regular, flexible and casual work.

We have shifts available in the Morning, Afternoons and Nightshifts and these are available Monday - Friday and Weekends.

Essential Criteria:

- Cert III Aged Care (Essential)
- Current Police Clearance or willing to obtain
- Current WA C Class Drivers Licence
- Reliable Transport
- Available to start ASAP

Fig. 10. The job advertisement before and after graphics and links removed

Text Processing Process

All the advertisements are loaded as a corpus language resource into the GATE system as shown in Figure 11.

Index	Document name
0	0.htm_00C89
1	1.htm_00C8A
2	10.htm_00C8B
3	100.htm_00C8C
4	101.htm_00C8D
5	102.htm_00C8E
6	103.htm_00C8F
7	104.htm_00C90
8	105.htm_00C91
9	106.htm_00C92
10	107.htm_00C93
11	11.htm_00C94
12	12.htm_00C95
13	13.htm_00C96
14	14.htm_00C97
15	15.htm_00C98
16	16.htm_00C99

Fig. 11. A collection of job advertisements loaded into corpus

Document reset process clears the results of any previous execution on a document and refreshes the system for the next processes. Tokenization process breaks up the text into tokens as shown in Figure 12 (left hand box). Then the sentence boundaries are determined as shown in Figure 12 (right hand box).

21 Mar 2013 Location: Bunbury & South West Salary: $20 - $24.99 per hour Work type: Part Time Classification: Healthcare & Medical Nursing - Aged Care Aged Carers $23+ Per Hour plus Allowances Facilities in Bunbury, Eaton and Harvey Regular shifts available Start ASAP!!	21 Mar 2013 Location: Bunbury & South West Salary: $20 - $24.99 per hour Work type: Part Time Classification: Healthcare & Medical Nursing - Aged Care Aged Carers $23+ Per Hour plus Allowances Facilities in Bunbury, Eaton and Harvey Regular shifts available Start ASAP!!

Fig. 12. The job advertisement is breaking into tokens and then being sentence split

The results of POS tagging and morphological analysis are shown in Figure 13. The results shown are for a sentence of "a leading Western Australian owned and managed company specialising in all aspect of the recruitment and labour hire.". POS tagging determines the usage of a word (noun, verb etc.) and the morphological analyser find the root word (e.g. manage for managed).

```
category=DT, kind=word, length=1, orth=lowercase, root=a, string=a}
affix=ing, category=VBG, kind=word, length=7, orth=lowercase, root=lead, string=leading}
category=NNP, kind=word, length=7, orth=upperInitial, root=western, string=Western}
category=JJ, kind=word, length=10, orth=upperInitial, root=australian, string=Australian}
affix=ed, category=VBD, kind=word, length=5, orth=lowercase, root=own, string=owned}
category=CC, kind=word, length=3, orth=lowercase, root=and, string=and}
affix=ed, category=VBN, kind=word, length=7, orth=lowercase, root=manage, string=managed}
category=NN, kind=word, length=7, orth=lowercase, root=company, string=company}
affix=ing, category=VBG, kind=word, length=12, orth=lowercase, root=specialise, string=specialising}
category=IN, kind=word, length=2, orth=lowercase, root=in, string=in}
category=DT, kind=word, length=3, orth=lowercase, root=all, string=all}
affix=s, category=NNS, kind=word, length=7, orth=lowercase, root=aspect, string=aspects}
category=IN, kind=word, length=2, orth=lowercase, root=of, string=of}
category=DT, kind=word, length=3, orth=lowercase, root=the, string=the}
category=NN, kind=word, length=11, orth=lowercase, root=recruitment, string=recruitment}
category=CC, kind=word, length=3, orth=lowercase, root=and, string=and}
category=NN, kind=word, length=6, orth=lowercase, root=labour, string=labour}
category=NN, kind=word, length=4, orth=lowercase, root=hire, string=hire}
category=., kind=punctuation, length=1, root=., string=.}
```

Fig. 13. POS and morphological checking of the job advertisement

Gazetteer and Annotation Processes

Recognition of concepts appearing in the text by the flexible gazetteer shows in Figure 14. The word-list based gazetteer is used to recognise salary amounts and dates appearing in the text. The information on amounts and dates is not available in EDO.

Fig. 14. The EDO based gazetteer recognises the concepts in the advertisement text

Ontology Population Update Process

The information extracted by the previous steps is then populated into EDO by creating a knowledge instance for the advertisement with properties shown in Figure 15.

Fig. 15. Extracted information is populated into EDO

6.2 SPARQL Queries

We detail the testing that was performed on the framework in this section. SPARQL queries have been run on EDIT to check that the tool produces the expected outcome and can, as such, be approved for user uptake in industry. Once the information about a set of advertisements is populated in the ontology by EDIT, the EDO can then be queried using SPARQL to get a range of quantitative data from the ontology. We choose SPARQL because it is a query language that can query ontologies and produce the desired summarised results which can be then used for EDIT validation.

The following shows how a variation of data has been filtered and retrieved from EDIT through the submission of SPARQL queries at the endpoint. To make the text more readable, the following prefixes are introduced:

owl: < http://www.w3.org/2002/07/owl#>
rdf: < http://www.w3.org/1999/02/22-rdf-syntax-ns#>
rdfs: < http://www.w3.org/2002/07/owl#>
edo:< http://www.semanticweb.org/ontologies/2011/7/EmployerDemandOntology.owl#>

Figure 16 indicates the query and subsequent results to retrieve all the jobs and their locations that were advertised on 18 March 2013.

```
                                  Query

PREFIX edo: <http://www.semanticweb.org/ontologies/2011/7/
EmployerDemandOntology.owl#>
SELECT *
                WHERE { ?job edo:hasLocation ?Location; edo:adDate ?adDate
Filter(?adDate= "18 Mar 2013" @en) }

ORDER BY ?Vacancy

                                  Result

1.htm        Perth          "18 Mar 2013"@en
10.htm       Perth          "18 Mar 2013"@en
17.htm       Perth          "18 Mar 2013"@en
56.htm       Yanchep        "18 Mar 2013"@en
56.htm       Perth          "18 Mar 2013"@en
60.htm       Perth          "18 Mar 2013"@en
60.htm       Geraldton      "18 Mar 2013"@en
81.htm       Perth          "18 Mar 2013"@en
94.htm       Perth          "18 Mar 2013"@en
94.htm       Bunbury        "18 Mar 2013"@en
97.htm       Perth          "18 Mar 2013"@en
```

Fig. 16. A query to retrieve all the job advertisements and their locations that were advertised on 18 March 2013

Figure 17 shows the query that retrieves all the instances of job advertisements that were advertised in each location in Western Australia, and the number of advertisements for each of these geographical locations.

```
                                  Query
PREFIX edo: <http://www.semanticweb.org/ontologies/2011/7/EmployerDemandOntology.owl#>

SELECT (SAMPLE(?location) AS ?location) (COUNT(?location) as ?number)

        WHERE {

        ?job edo:hasLocation ?location

        }
GROUP BY ?location
ORDER BY ?location


                                  Result

Albany              "8"^^<http://www.w3.org/2001/XMLSchema#integer>
Bunbury             "27"^^<http://www.w3.org/2001/XMLSchema#integer>
Busselton           "6"^^<http://www.w3.org/2001/XMLSchema#integer>
Esperance           "3"^^<http://www.w3.org/2001/XMLSchema#integer>
Geraldton           "4"^^<http://www.w3.org/2001/XMLSchema#integer>
Harvey              "2"^^<http://www.w3.org/2001/XMLSchema#integer>
Mandurah            "5"^^<http://www.w3.org/2001/XMLSchema#integer>
MargaretRiver       "1"^^<http://www.w3.org/2001/XMLSchema#integer>
Northam             "4"^^<http://www.w3.org/2001/XMLSchema#integer>
Perth               "74"^^<http://www.w3.org/2001/XMLSchema#integer>
Yanchep             "1"^^<http://www.w3.org/2001/XMLSchema#integer>
```

Fig. 17. A query to retrieve all the job advertisements that were advertised in the various locations in Western Australia

Figure 18 provides the query to show the number of job advertisements that had *OtherRequirements* listed, and what each of those requirements were.

Query

PREFIX edo: <http://www.semanticweb.org/ontologies/2011/7/EmployerDemandOntology.owl#>

SELECT (SAMPLE(?req) AS ?req) (COUNT(?req) as ?number)

     WHERE {

     ?job edo:hasOtherRequirements ?req

     }

GROUP BY ?req

Result

OwnTransport	"3"^^<http://www.w3.org/2001/XMLSchema#integer>
OwnVehicle	"3"^^<http://www.w3.org/2001/XMLSchema#integer>
PoliceClearance	"56"^^<http://www.w3.org/2001/XMLSchema#integer>
WorkingRights	"6"^^<http://www.w3.org/2001/XMLSchema#integer>
WorkingWithChildrenCheck	"1"^^<http://www.w3.org/2001/XMLSchema#integer>

Fig. 18. A query to establish the number of job advertisements that were advertised with other requirements

Table 1 provides a range of metrics that are relevant to EDIT, such as the number of job advertisements that were processed during the one month's trial period for the EDIT, and the number of instances that were populated that have the object properties *hasSalary*, *hasOtherRequirements*, and *hasLocation*.

Table 1: A range of metrics that were processed during trial period

Metric	Quantity
Number of job advertisements processed	105
Number of new data type properties created	2
Number of data type property instances asserted	131
Number of job advertisements with a date recorded	122
Number of new object properties created	6
Number of object properties asserted	495
Number of *hasRequirement* instances populated	2
Number of *hasSalary* instances populated	9
Number of *hasOtherRequirements* instances populated	69
Number of *hasLocation* instances populated	135
Number of *hasEmployeeQualificationLicenceOrRegistration* instances populated	35
Number of *hasAttributeRequirements* instances populated	244
Number of *hasEmployeeLanguage* instances populated	10

7. LIMITATION AND ONGOING DEVELOPMENTS

There are limitations in EDIT which needs improvement in future work. The SEEK Gazetteer, which enables EDIT to find the salary of a job and the date of publication of an advertisement, has a minimal functionality. Its functionality can be enhanced through extension of the gazetteer by adding more names into its lists and adding more JAPE grammar to the JAPE Transducer to process these lookups.

The approach followed in EDIT is to use a locally stored ontology, and it is consequently only able to extract information that has a knowledge representation in the ontology. The functionality of EDIT can be enhanced in future work: instead of using such an isolated ontology it can be linked with the semantic web to search for a wider range or concepts.

EDIT is based on a fully automated process that looks for the concepts in the job advertisements that match those contained within the EDO. Therefore, sometimes it

wrongly associates words which are used in the job advertisements' text in a different sense to the meaning implied by the concepts contained in EDO. These mistakes cannot be completely removed, but it can be reduced to some extent by enhancing the usage of the Part of Speech Tagger's output and incorporating more JAPE grammar.

The interlinking of EDO is another important work that needs to be done in the near future. Interlinking EDO with other relevant well-known ontologies encourages information interoperability. EDO can be used and adopted in the community to produce network effects. This was also highlighted in (Hepp, 2007): "ontologies exhibit positive network effects, such that their perceived utility increases with the number of people who commit to them which comes with wider usage". The Ontology Usage Analysis Framework (Ashraf, Hussain, & Hussain, 2014) has been developed to empirically analyze the use of ontologies and rank them based on their usage. Hence, uptake ontologies can be reused and adopted. For the EDO interlinking process, different vocabularies such as Friend-of-a-Friend (FOAF), Dublin Core (DC), Simple Knowledge Organization System (SKOS), and Semantically Interlinked Online Communities (SIOC) are all useful.

8. CONCLUSIONS

In this paper, we address the skill shortage issue as a real need for proper investigation and workforce development planning into the future. We review existing approaches to gathering employer-demand information. In general, those approaches have been manual to date and mainly ad-hoc, not founded on structured frameworks.. We then introduce the Employer Demand Intelligence Tool (EDIT) aiming to automatically populate instance knowledge extracting from job advertisements published online. Our approach of associating a domain concept with a terminology in the employer-demand domain is novel. It involves finding the appropriate Employer Demand Ontology (EDO) concept for each word of a terminological string found in the online job advertisements, including the appropriate conceptual relationships between the concept components. EDO dataset as outcome is used to identify current employer-demand needs analysed in real time from online sources. We verify our approach in a case study and validate the tool through a set of queries.

The employer-demand intelligence framework is an aspiring effort and much of its capacity has been discussed as well as its limitations and potential enhancements that need to be elucidated and marked as future work. This research project has shown that, while developing a comprehensive employer-demand intelligence framework is an ambitious endeavor, it nevertheless is possible when the framework is developed incrementally. That said, there are still several rather challenging areas in this research that need further attention.

REFERENCES

Access Economics. (2009). Employment forecasts.

Ashraf, J., Hussain, O. K., & Hussain, F. K. (2014). Making sense from Big RDF Data: OUSAF for measuring ontology usage. *Software: Practice and Experience*, n/a-n/a. doi: 10.1002/spe.2272

Biesalski, E., & Abecker, A. (2005). Human Resource Management with Ontologies. In K.-D. Althoff, A. Dengel, R. Bergmann, M. Nick & T. Roth-Berghofer (Eds.), *Professional Knowledge Management* (Vol. 3782, pp. 499-507): Springer Berlin Heidelberg.

Bizer, C., Heese, R., Mochol, M., Oldakowski, R., Tolksdorf, R., & Eckstein, R. (2005). The Impact of Semantic Web Technologies on Job Recruitment Processes. In O. Ferstl, E. Sinz, S. Eckert & T. Isselhorst (Eds.), *Wirtschaftsinformatik 2005* (pp. 1367-1381): Physica-Verlag HD.

Chamber of Minerals and Energy WA. (2009). Developing a Growth Outlook for WA's Minerals and Energy Industry. Perth.

Cunningham, H., Maynard, D., Bontcheva, K., Tablan, V., Aswani, N., Roberts, I., . . . Peters, W. (2013). Developing Language Processing Components with GATE Version 7 (a User Guide).   Retrieved

13 September, 2013, from http://gate.ac.uk/sale/tao/split.html

Curtain, R. (1988). *Addressing Skill Shortages: The Limitations of Macro-Level Labour Market Forecasting and an Alternative Approach.* Paper presented at the Australian Skills Shortages and Labour Market Data, Sydney.

DEEWR. (2008-09). Skill Shortage Methodology. Retrieved 15 April 2010, from http://www.workplace.gov.au/NR/rdonlyres/09321BDC-74EA-444A-B20D-9B3025DBC42B/0/WorkplaceMethodologyPaper.pdf

Dorn, J., Naz, T., & Pichlmair, M. (2007). *Ontology development for human resource management.* Paper presented at the 4th International Conference on Knowledge Management.

García-Sánchez, F., Martínez-Béjar, R., Contreras, L., Fernández-Breis, J. T., & Castellanos-Nieves, D. (2006). An ontology-based intelligent system for recruitment. *Expert Systems with Applications, 31*(2), 248-263. doi: 10.1016/j.eswa.2005.09.023

Gómez-Pérez, A., Ramírez, J., & Villazón-Terrazas, B. (2007). An Ontology for Modelling Human Resources Management Based on Standards

Knowledge-Based Intelligent Information and Engineering Systems. In B. Apolloni, R. Howlett & L. Jain (Eds.), (Vol. 4692, pp. 534-541): Springer Berlin / Heidelberg.

Green, F., Machin, S., & Wilkinson, D. (1998). The Meaning and Determinants of Skills Shortages. *Oxford Bulletin of Economics and Statistics, 60*(2), 165-187. doi: 10.1111/1468-0084.00093

Gualtieri, A., & Ruffolo, M. (2005). *An Ontology-Based Framework for Representing Organizational Knowledge.* Paper presented at the I-KNOW'05, Graz, Austria.

Hayton, G. (1988). *Translation of Labour Market Needs Into Training Responses.* Paper presented at the Australian Skills Shortages and Labour Market Data, Sydney.

Hepp, M. (2007). Possible ontologies: How reality constrains the development of relevant ontologies. *Internet Computing, 11*(1), 90-96.

JobsDB Dimension. (2011). JobsDB Dimension : Introduction. Retrieved 27 February, 2013, from http://www.jobsdb.com/Dimension/

Kumaran, V. S., & 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. doi: 10.1504/ijmso.2013.054184

Lau, T., & Sure, Y. (2002). Introducing Ontology-based Skills Management at a large Insurance Company. *IN PROCEEDINGS OF THE MODELLIERUNG 2002*, 123-134.

Lv, H., & Zhu, B. (2006, 17-19 Nov. 2006). *Skill ontology-based semantic model and its matching algorithm.* Paper presented at the Computer-Aided Industrial Design and Conceptual Design, 2006. CAIDCD '06. 7th International Conference on.

MacDonald, K., & Klinger, P. (2009, 17 Sep 2009). State told to build up trainees for boom, *The West Australian.*

O'Callaghan, A. (2008). Latest Trends in Recruitment & Selection. Johannesburg: FASSET.

Probyn, A. (2009, 2 Sep 2009). PM plans a blue-collar boom, *The West Australian.*

Sowa, J. F. (2009). Building, Sharing, and Merging Ontologies. Retrieved 10 April 2013, from http://www.jfsowa.com/ontology/ontoshar.htm

Stanimirović, A., Bogdanović, M., & Stoimenov, L. (2013). Methodology and intermediate layer for the automatic creation of ontology instances stored in relational databases. *Software: Practice and Experience, 43*(2), 129-152. doi: 10.1002/spe.2103

Storey, K. (2001). Fly-in/Fly-out and Fly-over: mining and regional development in Western Australia. *Australian Geographer, 32*(2), 133 - 148.

Terziev, I., Kiryakov, A., & Manov, D. (2005). Base upper-level ontology (BULO) Guidance. In S. S. E. K. Technologies (Ed.).

Trichet, F., & Leclère, M. (2003). A Framework for Building Competency-Based Systems Dedicated to Human Resource Management. In N. Zhong, Z. Raś, S. Tsumoto & E. Suzuki (Eds.), *Foundations of Intelligent Systems* (Vol. 2871, pp. 633-639): Springer Berlin Heidelberg.

Wibisono, A., Koning, R., Grosso, P., Belloum, A., Bubak, M., & de Laat, C. (2013). OIntEd: online ontology instance editor enabling a new approach to ontology development. *Software: Practice and Experience, 43*(11), 1319-1335. doi: 10.1002/spe.2141

Wilcock, G. (2009). Introduction to Linguistic Annotation and Text Analytics: Morgan & Claypool.

Wongthongtham, P. (2006). *A methodology for multi-site distributed software development*. Perth: Curtin University.

Zemmouchi-Ghomari, L. (2012). HERO Higher Education Reference Ontology. Retrieved 12 November, 2012, from http://herontology.esi.dz/content/downloads