# Automatic Extraction of Usable Information from Unstructured Resumes to Aid Search

Sunil Kumar Kopparapu
TCS Innovation Labs - Mumbai
Tata Consultancy Services
Thane (West), Maharastra 400 601, INDIA.
Email: SunilKumar.Kopparapu@TCS.Com

*Abstract*—**This paper describes a system for automated resume information extraction to support rapid resume search and management. The system is capable of extracting several important informative fields from a free format resume using a set of natural language processing (NLP) techniques. We describe a working system, for automatic resume management. The system is capable of extracting six major fields of information as defined by HR-XML[8]. Experimental results carried out on a large number of resumes show that the proposed system can handle a large variety of resumes in different document formats with a precision of 91% and a recall of 88%**

***Keywords-component; natural language processing; resume manager***

## I. BACKGROUND

Large enterprises and head-hunters ([5],[6]) receive several hundreds of resumes from job applicants every day. In general there is no standard format in which a resume can be written. To induce standards so that the resumes can be electronically cataloged and searched, enterprises force job seekers to fill an online template. While this process, helps the enterprise to effortlessly and quickly search for the right applicant, it induces unnecessary constraint on the applicant to fill in a different templates each time depending on the enterprise to which they are applying. A major problem associated with this approach is that the applicant is forced to tune their resume to match the style of the template which might not be able to capture all the details that the applicant might wish to display on their resume. Additionally, for the enterprise, the online template needs to be changed with time because of newer job descriptions or job types. Ideally, an enterprise would do away with forcing its applicants to fill in a predefined template provided they had access to a system that could extract the required information, both structured and unstructured, from any format of resume automatically. The benefit of such a system is that it would support automatic construction of an electronic resume database and would enable quick processing of resumes received by searching and routing resumes to appropriate destinations. Automatic extraction of information from resumes with high precision and recall is not an easy task essentially because of the non-standardization of resume structure. In spite of constituting a restricted domain, resumes can be written in multitude of formats (e.g. structured tables or plain texts) and in different file types (e.g. .txt, .pdf, .doc(x) etc.). Moreover, writing styles could be much diversified. Kun et al. [1] have proposed a resume information extraction using a cascaded hybrid model. It is a two pass approach; in the first pass, the resume is segmented into a set of consecutive labeled blocks which indicates the gross type of information that the block contains and in the second pass, the detailed information is extracted. Use of text mining for resume is mentioned in [3] and [4], while [7] mentions use of information extraction to process resume without much details. More recently the work has been on extracting special skills from resume to aid improve the precision with which a resume is selected [9] for high skilled jobs.

In this paper we describe a system, which is capable of processing resumes in multitude formats and forms and building an electronic database from the resume. Unlike [1] we propose a one pass system. The system aims to aid a large enterprise by removing the manual effort in screening resumes received by them to ascertain the suitability of candidate. The organization of the paper is as follows: In Section II., we describe the system that is capable of automatically extracting relevant information from a resume and pushing the information into a database. The extraction of relevant information is based on a set of natural language processing and pattern matching techniques described in detail in Section III. The complete system is web enabled to make it reachable to a large number of people within the company. In Section IV. we discuss the performance of the system in terms of precision and accuracy of the system and we conclude in Section V.

## II. SYSTEM DESCRIPTION

The system is a web based client-server which is capable of automatically extracting information from resumes in English language and populating a structured database. The complete system consists of several modules as depicted in Fig. 1. It comes with an interface (see Fig. 2 and 3) which allows for searching resumes populated in the database. The information extraction module is by and large the most significant component of the system. The information extraction module is capable of extracting important relevant information from a free format resume automatically. The database build module populates the database with the extracted information and builds a resume database. The current search module enables a user to search resumes with some particular criteria in the

**978-1-4244-6789-1/10/$26.00 ©2010 IEEE**

resume database. However a natural language interface to search resumes to enable searches like "Show me all the resumes that have more than 3 years of java experience" would be an ideal interface to have.

The input module is a web interface which allows input of resume to the system. The system, has no constraint on the resume style or structure. The input module is additionally capable of accepting multiple resumes in the form of a .zip, .tgz, .7z, .Z, .gz file.

The information extraction module is capable of extracting automatically from any given resume, information like, total experience, date-of-birth, passport number, email-id, skill set and qualification. The information extraction uses a bunch of natural language processing (NLP) algorithms to extract relevant information from a free English resume. The information extracted by the information extraction module is populated into a database by the database build module. The search module gives an interface to query the system for a specific resume. The user can query the resume database based on a combination of the following criteria (a) age of candidate, (b) qualification, (c) software skills and (d) previous experience. All the resumes matching the criteria are displayed with a summary of the selected resume. Further a hyperlink allows the user to view the complete resume in its original form.

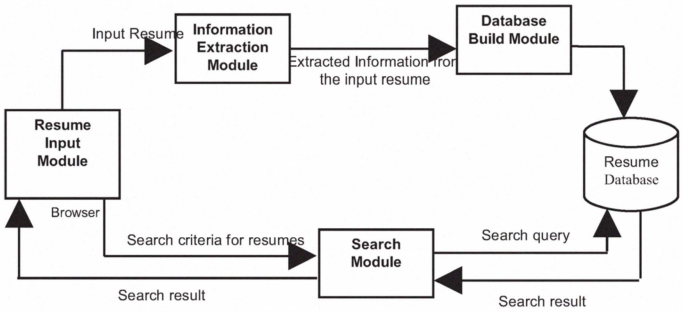


*Figure 1 : System Diagram of Resume Manager System.*

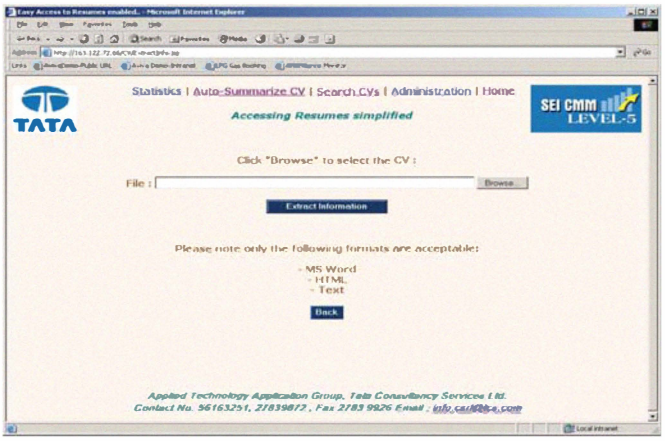


*Figure 2: Interface for information extraction from a resume.*

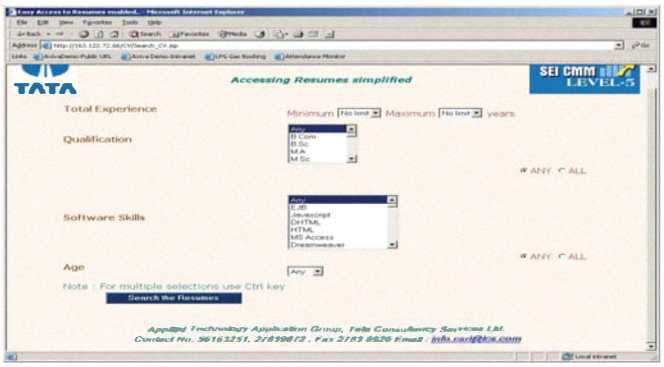


*Figure 3: The Search Interface to the resumes.*

Additionally, an administration module allows the configurability of the system to enable appending of new skill sets and deleting obsolete resumes etc. Also a routing module exists that can be configured to automatically send e-mail with the attached select resumes to desired managers in the organization.

## III. Information Extraction using NL techniques

Almost all resumes are unique in their structure and hence dissimilar, but one can assume a typical resume to have an overall hierarchical layered structure [2] as shown in Table 1 [1]). The first layer is composed of several general information blocks such as *personal information, education* etc. The second layer of structure is within the first layer and contains specific information corresponding to the layer 1. For example, the layer 1 *personal information* block consists of layer 2 information like *name, address and e-mail*. While this might not be true for all the resumes, the structure seems to be retained in the bulk of resumes. Additionally, the location of the information (like name, age etc) in resumes vary significantly from resume to resume. Our system can work on both layered structure and unstructured resumes.

*Table 1: Two layered structure of resumes*

| Information Hierarchy | | Information Type |
|---|---|---|
| General Info | | Personal Information; Education; Research Experience; Award; Activity; Interests; Skill |
| Detailed Info | Personal Detailed Info *(Personal Information)* | Name; Gender; Birthday; Address; Phone; Mobile; Email |
| | Educational Detailed Info *(Education)* | Graduation; Degree; University; Post Graduation |

Information extraction module is composed of several sub modules, each of which performs the task of extracting specific information. The main sub modules are (a) Qualification module, (b) Skill module (c) Experience module and (d) personal information extraction. While the qualification extraction sub-module extracts the graduating university name, degree and the class obtained. The skills extraction module extracts the skills of the candidate. Experience extraction module is capable or extracting the total experience, even when this information is not explicitly mentioned in the resume of the candidate. The name extraction module extracts applicant name and other information like date-of-birth, email-id and passport number. The extraction process uses a set of language processing techniques which are part heuristics and part pattern matching.

### *A. Qualification Extraction Module*

The resume is scanned to check the presence of qualification related keywords from the reference data files. Data files are created offline using a bunch of reference resumes in a semi-automatic manner. The knowledge-base related to qualification consists of a set of all likely qualifications in all possible morphological forms (example Bachelor of Arts, BA refer to the same thing). Typical entries are name of the qualification, university name and degree class. This module extracts the information and tags them accordingly. A typical output of this module would return B.E (First Class), M.E (IISC - Distinction). Note that if there are multiple qualifications specified in the resume, all the qualifications are extracted.

### *B. Skills Extraction Module*

Software skill extraction module aids search for skill sets listed in the knowledge base. The system is capable of taking care of different forms of skill sets, namely, synonyms and morphological forms of the skills. Additionally, like in all other modules, the system uses a spell checker to identify and resolve typographically errors in the resume. The procedure adopted by the skill extraction module is to initially form n-grams[1] ($n < 7$) from the resume. For example, if the resume contained text "*Project Environment: Informatica, Cognos Impromptu Administrator, Cognos Powerplay, Cognos Scheduler….*" then the following n-grams would be constructed, namely, *Project, Environment, Project_Environment, Informatica, Cognos, Impromptu, Administrator, Cognos_Impromptu, Impromptu_Administrator Cognos_Impromptu_Administrator, Cognos, Powerplay, Cognos_Powerplay, Cognos, Scheduler, Cognos_Scheduler*. Then each of these n-grams is pattern matched with the skill set in the knowledge base. If a match is found it is marked as a skill set. This processing is done for the whole of the document by processing each and every n-gram until the end of the document is reached, the largest n-gram matching the knowledge base is output as the skill set.

### *C. Experience extraction Module*

Experience of a candidate is one of the most important fields that is used for identifying candidate for the job. The experience extraction module tries to identify if the candidate has mentioned his total experience like in "I have a total experience of 7 years in the IT industry" by looking for patterns like "total years of experience", "experience total of ….years", "entry level experience", "years of professional experience", "years of experience", "professional exposure of….years/months" etc.. If there is no mention of total number of years of experience in a resume, then the experience extraction module identifies all the time periods spent by the candidate in different projects and then sums up the experience in each project to come out with a total number of years of experience.

If total experience is not extracted then the resume is scanned to identify a pattern which has two sets of numbers to represent month and / or year. If present, it extracts both the month and / or year from the line, calculates the time period in between the two dates and stores it. Similarly it sums up all the time duration to calculate the total experience of the candidate and ultimately post-processing has been done to convert the time duration (which can be in different forms, namely, "4 years 2 months", "50 months" etc) into a pre-determined format.

### *D. Name extraction Module*

Name extraction collects all possible name candidates and determines the word (or set of words) with highest probability as the candidate name. The processing involves identification of all parts of the resume having the pattern "*name*" but no "*project*" or "*father*" patterns preceding the pattern "*name*" (to avoid project name and fathers name which are common in a resume). The isolated region (typically a line of text) is segmented into units using ':', ':-', '-', ',', '_' as the delimiter. The non-dictionary word in the segmented line becomes a candidate for the name of the person. In the event no name is

[1] N-gram is a group of n words which are clubbed together and processed as a single pattern. Usually in natural language processing bi-gram and tri-grams are used for processing.

found; the module checks if the first 5 lines of the resume contain the name followed by a check on the last line of the resume. A name candidate extracted by the module is cross checked with the name of the file, if pattern extracted as name is part of the filename, then the weightage for that particular pattern as a name candidate is increased. In all the above processing the name check is performed based on certain assumptions like (a) a word present in the dictionary, (b) a word having a number or a special characters or (c) the length of the word is excessive of 20 characters do not qualify as candidate name.

### *E. Other*

#### *1) Date-of-birth (DOB) field extraction*

The resume is scanned and all the line with the keywords patterns like, '*birth*', '*born*', '*dob*' with the exception of those lines having the pattern '*place*' are isolated and a search for date pattern is done. Following are the examples of date format that the system recognizes (a) DD[/.- ]MM[/.- ]YYYY, (b) DD[/.- ]MM[/.- ]YY, (c) DD[th/st] MMM YYYY, (d) DD[/.- ] MMM[/.- ]YYYY. In addition, all the morphological forms of days (09, 9) and months (January, Jan) are supported. The identified pattern is then post processed to translate the identified DOB into a predetermined format in the form DD/MM/YYYY and then calculates the age of the candidate by comparing the DOB with the current date obtained from the system.

#### *2) Email Extraction Module*

Any line having the patterns like "@", "[at]" become candidate for email-id of the candidate. This mechanism however determines all the email addresses in the resume. A post processing is done by analyzing all the email addresses returned by the system to pick the probable email address. For example, all email addresses collected from the "References" part of the resume are not considered as the possible email id of the candidate. Email id having the candidates name embedded in it can be checked to get best possible candidate email.

## IV. Experimental Results

The performance of the system, as is applicable to all natural language processing based systems, depends on the accuracy and span of the knowledge base. Knowledge base is, as mentioned earlier, is built using several reference resumes in a semi-automatic process. It is thus important to measure the performance of the system on a set of resumes that have not been used as reference resumes to build the knowledge base. A total of 100 resumes, mostly of the candidates who were applying to a company, were obtained. A set of 50 resumes, picked randomly, were used to populate the knowledge base. These 50 resumes formed the set of reference resumes. The performance of the system was tested on the remaining 50 resumes. The criteria used for evaluating the performance of the system are precision (p), and recall (r). These metrics are used to measure the performance of any information extraction (IE) systems. Precision and recall metrics are defined as follows:

$$p = \frac{Number\ of\ correctly\ extracted\ fields\ by\ system}{Total\ number\ of\ fileds\ extracted\ by\ system}$$

$$r = \frac{Number\ of\ correctly\ extracted\ fields\ by\ system}{Actual\ numberof\ fields}$$

Note that in the above definition, a field is generic and could be any information (name, date-of-birth, qualification, etc) extracted from the resume. Also note that in the definition of recall, the denominator requires the 'actual' correct fields present in the resume. To obtain this information, all the 50 test resumes were manually annotated with correct fields. This information, however was not used by the system.

| | |
|---|---|
| Name : | Chandra Sekhar Gajula |
| Software Skills : | Informatica, Cognos, Powercenter, C, Unix, MS Office, Pl/SQL, D2K, Oracle, Informix, Toad, Erwin, Cognos power play, Cognos scheduler, Cognos web query, Cognos iwr, Cognos upfront, Business Objects, Cognos Impromptu, Cognos transformer, ERP, Visual basic |
| Qualification : | B.Com(Venkateswara University), M.B.A, P.G.D.H.R.M, Diploma |
| Experience : | More than 3 years |
| Email : | chandugajula@yahoo.co.uk |

*Figure 4: Typical outputs of resume Manager.*

Fig. 4 shows typical outputs of the system. Specifically it is the output produced by the system. Note that, there are a total of 17 fields that have been extracted by the system. These fields are *Biswajit Sarkar, Oracle, SQL, Pl/SQL, D2K, C, C++, Java, Unix, Windows, B.E (First Class), M.E (IISC - Distinction), M.Tech (BIT - Distinction), 18 years 1 month, sarkar_b1@rediffmail.com, sarkar_biswa@yahoo.com,* and *01/01/1965.*

The performance of the system was first tested with the train data (the 50 resumes used to create the knowledge base). The precision is about 91% and recall is about 88%. In the testing phase, with 50 resumes that were not used by the system to construct the knowledge base, the system overall performance was 87% on the precision scale and 71% on the recall metric. Fig. 5 shows the precision (on the x-axis) and recall (on the y-axis) of the individual resumes. Notice that all the resumes sit in the upper right hand corner in the plot indicating that the fields in all the resumes were extracted with high precision and high recall. Also notice that the precision-recall value of the train data is better than the precision-recall value of the test data. As expected, a down-left (in the x and y directions respectively) shift of the test data cluster (represented by ▲'s) with respect to the cluster for the train data (represented by ♦'s) is quite prominent

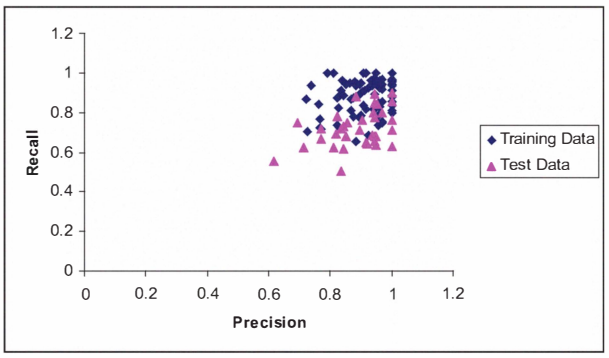


*Figure 5 Precision and recall plot for train dataset (♦) and test datasets (▲).*

## V. Conclusions

With increasing use of electronic media to seek jobs and fill vacancies, there is a need for a tool which is capable of extracting useful information from a free form resumes. In this paper, we have proposed a functional and automatic information extraction tool for both structured and unstructured resumes to aid electronic search. A mix of natural language processing techniques and heuristics were used to build information extraction modules to aid extraction of useful information from resumes. The knowledge base was created using reference resumes and the system was tested on a large number of resumes which was not part of the reference resumes. The current implementation extracts six major fields of information as defined by HR-XML [8], other relevant fields that can be extracted from the resume are contact telephone number, postal address, languages known, present company, designation.

## Acknowledgment

The authors would like to thank several past and current members of the TCS Innovation Lab - Mumbai, Tata Consultancy Services, for their active contributions during various phases of this work. Specifically, Prof PVS Rao, Akhilesh Srivastava and Sumitra Das.

## References

[1] Yu, K., Guan, G., and Zhou, M. "Resume Information Extraction with Cascaded Hybrid Model", in Proceedings of the 43rd Annual Meeting of the ACL, pages 499–506, Ann Arbor, June 2005.

[2] Finn, A. and Kushmerick, N. "Multi-level boundary classification for information extraction," in Proceedings of 15th European Conference on Machine Learning, Pisa, Italy, September 20-24, 2004.

[3] Nahm, U. Y. and Mooney, R. J. "Text mining with information extraction," in Proceedings of the AAAI 2002 Spring Symposium on Mining Answers from Texts and Knowledge Bases, pp. 60-67, Stanford, CA, March 2002.

[4] Makhoul, J., Kubala, F., Schwartz, R. and Weischedel, R. "Performance measures for information extraction," in Proceedings of DARPA Broadcast News Workshop, (Herndon, VA), February 1999.

[5] Monster, http://www.monster.com

[6] Naukari, http://www.naukari.com

[7] REX: http://www.resumemirror.com/products/resume-processing.html

[8] HR-XML, http://www.hr-xml.org/hr-xml/wms/hr-xml-1-org/index.php?language=2

[9] Maheshwari, S., Sainani, A. and Reddy, P. K. "An Approach to Extract Special Skills to Improve the Performance of Resume Selection," in 6th International Workshop on Databases in Networked Information Systems (DNIS2010), LNCS 5999, pp. 256 273, 2010.