Web-based recruiting

Framework for CV structuring

Soumaya Amdouni
University of Tunis
Institut Supérieur de Gestion
Cité Bouchoucha, Le Bardo, TUNISIA
Soumaya.miagiste@gmail.com

Wahiba Ben abdessalem Karaa
University of Tunis
Institut Supérieur de Gestion
Cité Bouchoucha, Le Bardo, TUNISIA
wahiba.Abdessalem@isg.rnu.tn

Abstract—**Recently, Information and Communication Technologies have introduced new practices in human resource management functions such as e-recruitment. Job seekers submit their Curriculum Vitae (CV) via the Web, or send them directly to a company. The area of e-recruitment is facing a growing number of these documents which are in different formats, and contain a large amount of information. Then, it has become imperative to use automated techniques to identify, extract, and exploit information from CVs to find the most appropriate one for a given post.**

Our work focuses on CVs analysis. We present a system for analyzing and structuring CVs which are in French language. For his end, we made an extension of General Architecture of Text Engineering (GATE) by formulating necessary rules that generate new annotations. The goal is to normalize the CV content according to the structure adopted by Europass CV. This action is guided by the HR-XML standard. An empirical study is conducted to validate the proposed process and we show that there is an improvement in the extraction phase.

Keywords-E-recruitment; CV; annotations; extraction; structuring; HR-XML; Europass CV.

I. INTRODUCTION

Computer Economics [www.computereconomics.com] estimates the total number of Internet users globally to be approximately 1.25 billion. This number represents a worldwide Internet penetration level of roughly 17% (based on a global population estimate of 6.57 billion according to the U.S. Census Bureau as of February, 2007). The revolution brought by ICTs has led to an acceleration and enlargement of information flows, circulating in companies. This information, which most of the time is text documents, has to be manipulated, classified, retrieved, etc. The growth of job market has also proven that traditional methods of recruitment are becoming inefficient [14]. The internet technology has radically changed the process of human resources management and has become an effective communication mean [11].

The idea behind e-recruitment is simple. Potential candidates postulate their CVs in a database on the Web where organizations can then search through thousands of CVs of persons looking for job that matches with their qualifications [12] [5]. So, CV analysis is an essential function in the technology of e-recruitment because the recruiter has to find the most appropriate candidate. However, there are different problems impeding the e-recruitment process at several levels. It's difficult to manage the large number of CVs that come in various forms via the web. Those troubles may lead to exclude interesting candidates, due to the multiplicity of broadcast media (email, CV databases…).

Structuring, extraction and integration of CVs information are important activities in the recruitment process. Indeed, the information retrieval in CVs, depends on content modeling, which aims to assign a semantic to improve the quality of retrieval results. Text mining is a new and exciting research area that tries to solve the information overload problem by using techniques such as machine learning, Natural Language Processing (NLP), Information Retrieval (IR), and knowledge management [10].

In this paper, we propose an approach based on NLP techniques to automate the process of e-recruitment. The main idea is to model the semantic content of unstructured CVs which are in different formats (PDF, doc, RTD, eml, etc.) following the structure of the Europass CV (http://europass.cedefop.europa.eu/) and using the GATE API (http://gate.ac.uk/).

The paper is organized into five sections. Section 2 presents some related works. Section3 gives an overview of GATE API and discusses our approach. Section 4 gives the validation results and the final section presents a conclusion.

II. RELATED WORKS

The CV is a semi structured document. We can identify the profile and skills of the candidate, by filling usually five parts (personal information, training, experiences, skills and other information). However, this document has often been developed over time, and some parts have been changed removed or added [4]. In addition CVs come to organizations in different formats (pdf, doc, rtf, ps, eml…) and the content is highly symbolic due to the use of acronyms. However, its semantic content is very dense, such as the description of different experiences, and level skills. The study of the more relevant document the CV to use it automatically has been a subject of many researches.

A project based on BONOM system [7], was developed to look for an adequate profile in a set of CVs. Agents are organized in field hierarchy according to the domain. Requests of user agents are conveyed through this hierarchy to specialized agent's sites. The system contains two phases: information extraction, and indexation.

Berio [3] was interested on applying knowledge techniques to competence extraction, and management. He used ontology techniques, e-learning system and CRAI model (Competency Resource Aspect Individual) to provide a representation of competencies required and acquired.

[9] outlines a HR-XML based prototype dedicated to the job search function. The system selects important information such as pay-check, topic, abilities, etc.

Moreover, another approach in e-recruitment domain [2] used Finite State Transducer formalism to extract key information from CV. This work proposed a model for the representation of the CV's content, and used Finite State Transducer formalism for this representation. So, the parser browses the CV in the XML format and identifies the different tags which should be in relation with employer requirements to construct the transducer, ensuring an easy and efficient CV retrieval.

Upon reading all the works cited above, we find that they focus on the axis of indexing and semantic annotation of documents after an information extraction.

In the context of semantic annotation, the main idea is to allow the person to annotate its document following a logic based on competencies such as CommonCV project [13] and the approach done by Berio [3]. The major lack in this work is the incompleteness of the given descriptions and the level of detail. On the other hand, criteria such as personal information: ages, sex, residence ... as well as diplomas and professional experience are required by some recruiters were not considered. In the context of semantic indexing, this technique does not enrich the documents with their semantic content; it just combines the concepts of used ontologies [7]. We also note that the techniques of Natural Language Processing still difficult to control. In the context of information extraction, the use of finite state transducer is a good technique to extract important CV content which facilitates the retrieval phase [2] however the author has just used an XML corpus which follows the Europass structure and ignored all other format of CVs.

In this context, we develop a prototype of an application to treat carefully the CVs in order to facilitate the e-recruitment process. The application can handle different formats of unstructured CVs by transforming them to structured XML documents following a standard structure.

III. The Approach

The CV structuring consists in identifying key information in this document. Thus, linguistic analysis is necessary to extract all objects such as name, sex, birth date, diploma, etc. In our work, we use GATE API and we extended it by new JAPE rules to extract information from CVs. Then we structure the extracted information according to the Europass CV following HR-XML standard.

The Europass CV is a formulary set up at European level that permits a chronological, systematic and dynamic presentation of individual's education, his qualifications and competencies.

HR-XML (Human Resource XML is a standard to describe vocabulary, specifically dedicated to the management in the company (www.hr-xml.org). It opens the way to publication of structured data. It is a set of XML specifications aimed at facilitating the exchange and automated processing of information related to human resources management.

A. GATE overview

Our system uses GATE API [6]. It is developed by the University of Sheffield. GATE is a framework and graphical development environment, which enables users to develop and deploy language engineering components and resources in a robust fashion. GATE contains different modules to process text documents. GATE supports a variety of formats (doc, pdf, xml, html, rtf, email…) and multilingual data processing using Unicode as its default text encoding.

To analyze CVs document we use the information extraction tool **ANNIE plugin** (**A** Nearly-New **IE** system) (Figure 1). ANNIE consists of pipelined components including Tokeniser, Gazetteer (system of lexicons), Pos Tagger, Sentence Splitter, Named Entity Transducer, and OrthoMatcher.

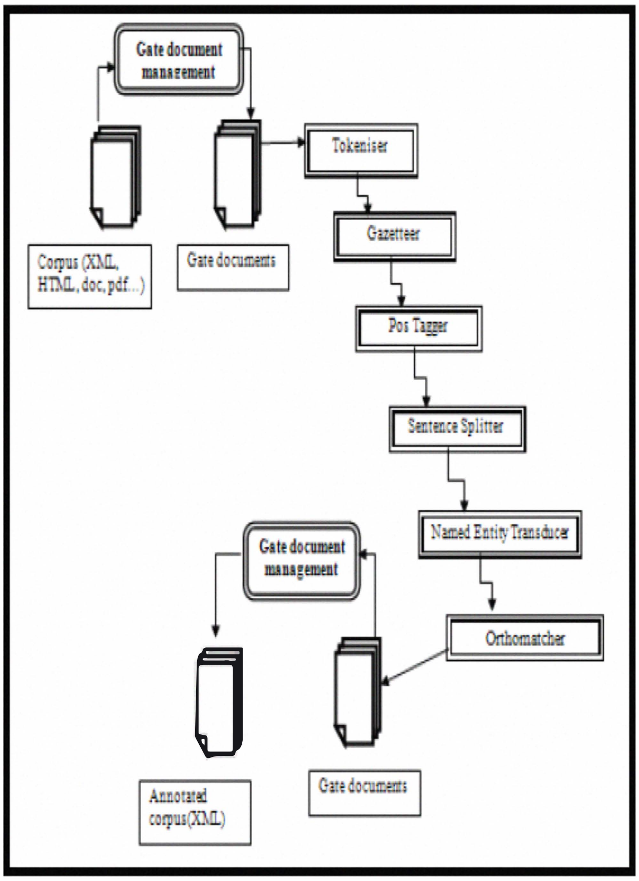


Figure 1. The ANNIE components in GATE.

-Tokeniser: The ANNIE Tokeniser identifies various symbols in texts (punctuation, numbers, symbols and different types). Tokenisers apply basic rules to input text in an attempt to identify these textual objects.

-Gazetteer: it is the component that creates annotation to provide information about entities such as persons, organizations, job titles, etc using lookup lists with one entry per line.

-Pos Tagger: GATE uses the Brill-style POS tagger, this component produces a tag to each word or symbol.

-Sentence Splitter: it is a module which identifies and annotates the beginning and end boundaries of each sentence. This module is required for the tagger.

-Named Entity Transducer: the NE transducer applies JAPE rules to the produced annotations to generate new annotations (Alani et al., 2003).

-OrthoMatcher: The OrthoMatcher is the module which performs coreference, or entity tracking, by recognizing relations between entities. In addition, it assigns annotations to previously unclassified names, based on relations with existing entities.

B. System architecture

The present framework uses the following components: Tokeniser, Gazetteer, Sentence Splitter and Named Entity Transducer. The entity recognition is the most important task, that's why, we extended GATE with additional extraction rules and additional Gazetteer lists to identify relevant entities in CVs such as name, nationality, address, experience, training, etc. In this task we used a learning corpus containing 50 CVs in French language and which are in different domains such as computer science, biology, medicine, linguistics, etc. This corpus helps us to write JAPE rules. We identify factors that allow us to locate pertinent information. We try to find a model in our records, trying for example to see that the information we want is always surrounded by the same words, or abutting a word of a specific grammatical category. The steps of our framework are as follows (Figure 2):

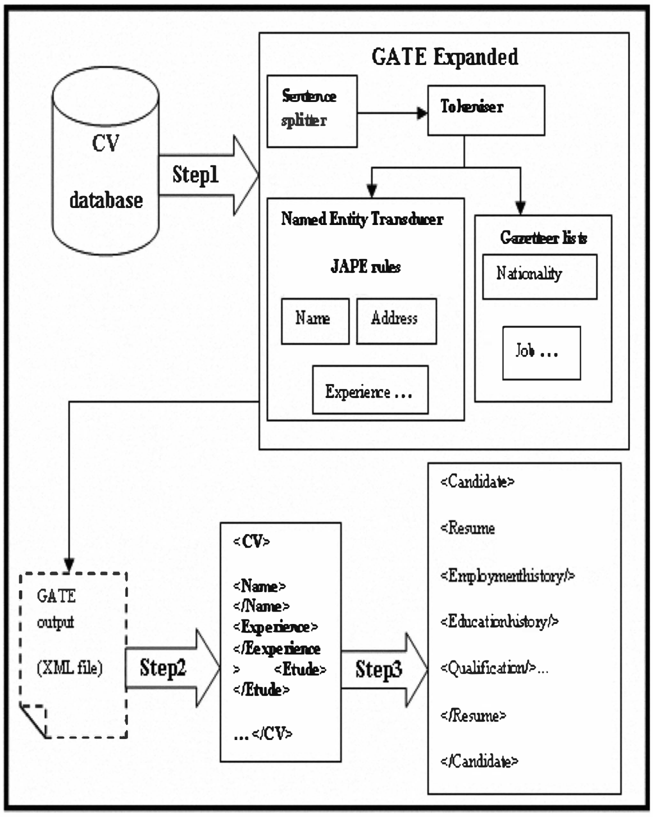


Figure 2. System architecture.

In a first step, the ANNIE component begins by identifying sentences in the processed CVs using Sentence Splitter component. The sentences are identified using annotations generated from the Sentence Splitter. For each sentence, a data structure is prepared (Tokeniser). The Tokeniser splits the text into very simple tokens such as numbers, punctuation and words of different types. The aim is to limit the work of the Tokeniser to maximize efficiency, and enable greater flexibility by placing the burden on the grammar rules, which are more adaptable. For example, we distinguish between words in uppercase and lowercase, and between certain types of punctuation. The named entities in the sentence are identified using annotations person, place, organization… that are already generated from the Named Entity Transducer sub component. The Named Entity Transducer operates with texts and produces information about the latter. It is based on reference annotation model. The reference model stores annotations in annotation graphs and this information is not embedded in the original text but instead the generated annotations refer to fragments of the text by reference. A GATE annotation consists of ID which is unique in the document, type which denotes the type of the annotation, start and end node, and a set of features which provides additional information. In the end of this first step, we generate an XML file containing all annotations. Then, during the second step we clean GATE output by removing unnecessary tags such as <sentence>, <token>... Finally during the final step, we check the integrity of extracted information to a specific grammar: the HR-XML schema related to the Europass CV.

The named entity recognition is the most important task in the information extraction process. GATE is a rich API with Jape rules and Gazetteer lists. However, these default rules and lists are insufficient to extract all the key information from CV. For this reason, we added to GATE extraction rules and lists to allow the identification of other types of entities. The GATE extension concerns mainly:

JAPE rules: Gender rules, name rule, phone rule, address rule, nationality rule, training rule, experience rule, organization rule, skills rule.

Gazetteer lists: nationality list, job list, organization list.

All new rules are integrated in GATE and tested in our CV corpus. Figure 3 is an example of JAPE rule to extract phone number. For this type information, it identifies elements of the form number.number.number or number or +number or (+number) number. Figure 4 is the JAPE rule to extract candidate gender which identifies elements of the form token that can be “homme” or “male” or “masculin” or “femme” or “femelle” or “féminin”. Figure 5 shows the JAPE rule to detect candidate name. The candidate's name is placed at the beginning of the CV and it contains less or equal than three words which have uppercase letters. Usually the letters of the last name is all capitals. We formulate this rule with the help of these remarks.

```
Rule:Phone1
Priority: 50
({Token.kind == number,Token.length >= 8}
|
(({Token.string=="("})?
({Token.string=="+"})?
{Token.kind == number,Token.length <= 3}
({Token.string==")"})?
({Token.string=="."})?
{Token.kind == number,Token.length <= 3}
({Token.string=="."})?
{Token.kind == number,Token.length <= 3}
({Token.string=="."})?
{Token.kind == number,Token.length <= 3}
({Token.string=="."})?
{Token.kind == number,Token.length <= 3}
))
:phone -->
  :phone.Phone= {kind = "phone", rule = "Phone1"}
```

Figure 3. Phone JAPE rule.

```
Phase:    Genre
Input:  Token
Options: control = appelt
Rule:Genre1
Priority: 50
( ( {Token.string == "homme"})|
({Token.string == "male"})|
({Token.string == "masculin"})|
({Token.string == "femme"})|
({Token.string == "female"})|

({Token.string == "féminin"})|
({Token.string == "Homme"})|
( {Token.string == "Male"})|
( {Token.string == "Masculin"})|
({Token.string == "Femme"})|
({Token.string == "Female"})|
({Token.string == "Féminin"}))
:genre -->
:genre.Genre= {kind = "genre", rule = "Genre1"}
```

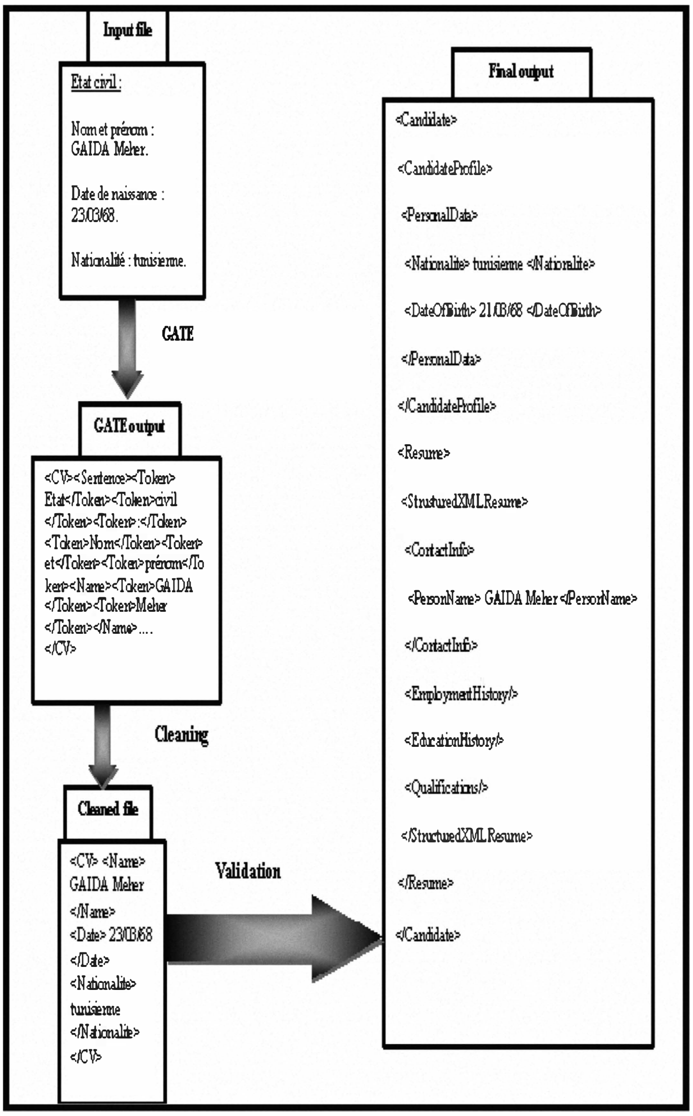
Figure 4. Gender JAPE rule.

```
Phase: Name
Input: Token
Options: control = appelt
// Name rule
Rule:Name1
(
{Token.orth == allCaps,Token.length
>=3,!Token.string =~ "[CURRICULUM]"}
({Token.orth == upperInitial,Token.length
>=3,!Token.string =~ "[VITAE]"}|{Token.orth ==
allCaps,Token.length >=2,!Token.string =~
"[VITAE]",!Token.string =~ "[Vitae]"})
({Token.orth == upperInitial,Token.length >=3})?
)
:name -->
:name.Name= {kind = "name", rule = "Name1"}
Rule:Name2
(
{Token.orth == upperInitial,Token.length
>=3,!Token.string =~ "Curriculum",!Token.string
=~ "Vitae"}
({Token.orth == upperInitial,Token.length
>=3,!Token.string =~ "Vitae"}| {Token.orth ==
allCaps,Token.length >=2})
({Token.orth == allCaps,Token.length >=3})?
)
:name -->
  :name.Name= {kind = "name", rule = "Namee2"}
```

Figure 5. Name JAPE rule.

IV. TEST AND VALIDATION

We have formed our CV corpus from a Tunisian recruitment firm (www.emploi.nat.tn) which is different from the learning corpus. It contains 150 CVs in different domains (biology, computer science, medicine, linguistic…). Then, we have tested our system on this corpus. The following example (Figure 6) shows the whole process. For a sample of CVs we applied GATE which generates an XML file containing all tags. We clean the file by removing unnecessary tags like <sentence>, <token> ... Finally; we structure the cleaned file according to the structure adopted by the Europass CV.



```
Etat civil :

Nom et prénom :
GAIDA Meher.

Date de naissance :
23/03/68.

Nationalité : tunisienne.
```

```
<CV><Sentence><Token>
Etat</Token><Token>civil
</Token><Token>:</Token>
<Token>Nom</Token><Token>
et</Token><Token>prénom</To
ken><Name><Token>GAIDA
</Token><Token>Meher
</Token></Name>....
</CV>
```

```
<CV> <Name>
GAIDA Meher
</Name>
<Date> 23/03/68
</Date>
<Nationalite>
tunisienne
</Nationalite>
</CV>
```

```
<Candidate>
<CandidateProfile>
 <PersonalData>
  <Nationalite> tunisienne </Nationalite>
  <DateOfBirth> 21/03/68 </DateOfBirth>
 </PersonalData>
</CandidateProfile>
<Resume>
 <StructuredXMLResume>
  <ContactInfo>
   <PersonName> GAIDA Meher </PersonName>
  </ContactInfo>
  <EmploymentHistory/>
  <EducationHistory/>
  <Qualifications/>
 </StructuredXMLResume>
</Resume>
</Candidate>
```

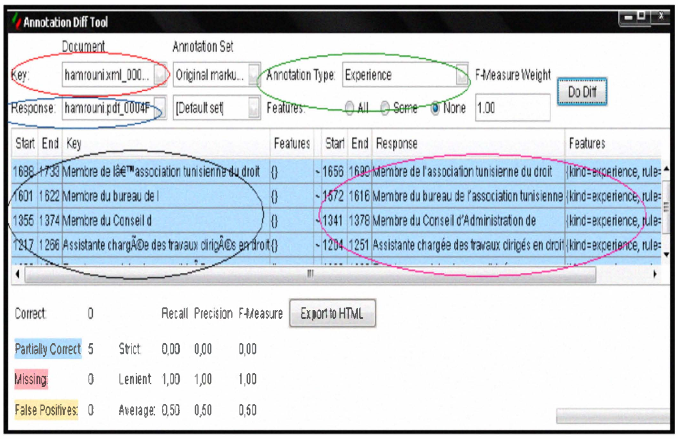
Figure 6. Example illustration.

The table below (Table I) represents the response time for 1, 10, 50 and total CV corpus during GATE processing and validation processing. Indeed, we wanted to evaluate the influence of response time compared to the size of the corpus thus we conclude that the response time is not too influenced by the corpus. For example a corpus of 10 CVs does not take 10*23s (23s is the response time for one CV).

TABLE I. RESPONSE TIME

CV number	Response time GATE	Response time Cleaning and validation
1 CV	23 s	15s
10CV	1mn 26s	1mn01s
50CV	2mn 45s	1mn54s
Total corpus (150CV)	5mn 9s	3mn14s

The system performance is determinate from the CV set using the AnnotationDiff tool [8] which compares the manual results with the system's results. Figure 7 shows a part of the AnnotationDiff viewer. The key document "hamrouni.xml" represents the hand annotated document and the response document "hamrouni.pdf" is the GATE document. So the Annotation Diff Tool will compare these two documents annotation by annotation for example in the figure the comparison concerns "experience" annotation. After that, the left information represent experience annotations extracted by the system and the other information are experience annotations from the document that we annotated manually. We have 5 partially correct annotations so the performance metrics Recall, Precision and F-measure are (0, 0, 0) when we consider that the partially correct are false and (1, 1, 1) when we accorded to partially correct answers.



Document: Key: hamrouni.xml_000... Response: hamrouni.pdf_0004F

Annotation Set: Original marku... [Default set]

Annotation Type: Experience

Features: ○ All ○ Some ● None

F-Measure Weight: 1.00

Start	End	Key	Features		Start	End	Response	Features
1688	1733	Membre de lâ€™association tunisienne du droit	{}	~	1656	1699	Membre de l'association tunisienne du droit	{kind=experience, rule=
1601	1622	Membre du bureau de l	{}	~	1572	1616	Membre du bureau de l'association tunisienne	{kind=experience, rule=
1355	1374	Membre du Conseil d	{}	~	1341	1378	Membre du Conseil d'Administration de	{kind=experience, rule=
1217	1266	Assistante chargÃ©e des travaux dirigÃ©s en droit	{}	~	1204	1251	Assistante chargée des travaux dirigés en droit	{kind=experience, rule=

			Recall	Precision	F-Measure
Correct:	0				
Partially Correct	5	Strict:	0,00	0,00	0,00
Missing:	0	Lenient:	1,00	1,00	1,00
False Positives:	0	Average:	0,50	0,50	0,50

Figure 7. Part of the AnnotationDiff viewer.

The corpus is evaluated according to precision, recall and F-measure with a half weight accorded to partially correct answers. The first table (Table II) shows the performance metrics for personal information part and the second table (Table III) shows it for experience, training and skills parts.

We calculate the performance metrics for each CV and the results given in tables are the averages of all measures for all CV database.

TTABLE II. PERFORMANCE METRICS FOR PERSONAL INFORMATION ANNOTATIONS

	Name	Gender	Email	Phone	Nationality	Address	Date
Recall	0.74	0.94	0.86	0.97	1	0.86	0.9
Precision	1	0.83	0.91	0.97	0.84	0.81	0.9
F-measure	0.85	0.88	0.88	0.97	0.91	0,83	0.9

TABLE III. PERFORMANCE METRICS FOR EXPERIENCE, TRAINING, AND SKILLS ANNOTATIONS

	Training	Institute	Experience	Language	Prog.lang	OS	Tool
Recall	0.83	0.85	0.81	0.88	0.89	0.97	0.97
Precision	0.87	0.74	0.85	0.82	0.89	0.98	0.97
F-measure	0.84	0.79	0.82	0.84	0.89	0.97	0.97

The results of our evaluation phase are satisfactory. The recall varies between 0.74 and 1, the precision between 0.74 and 1 and the F-measure between 0.79 and 0.97. So, we notice that the JAPE rules and Gazetteer lists are powerful. In addition, according to the table above we deduct that that competencies information and personal information have the best performance metrics because these information are easier to extract better than experience information or training information which are more ambiguous.

V. CONCLUSION

The goal with a semantic Web is to facilitate the realization of systems able to process knowledge, using processes similar to human reasoning, thereby obtaining more meaningful results and facilitating automated information and research by computers.

Our framework represents an automatic tool of CV analysis. It is an example of text mining in Human Resource Management area. It extracts heterogeneous information, organizes, and stores it as corpus XML following Europass CV structure.

In a future work we propose to improve the approach by CV classification according to employer's requirements. We have also proposed to analyze job offers and add a module which will be able to automatically match job offers to CVs.

ACKNOWLEDGMENT

The authors would like to thank Laboratory of Operations Research, Decision and Control of Processes Tunisia (LARODEC) for its support, and Gate users for their disponibilities and their helps.

REFERENCES

[1] Alani, H., Kim, S., Millard, D.E., Weal, M.J., Hall, W., Lewis, P.H., and Shadbolt, N.R. (2003). Automatic Ontology-based Knowledge Extraction from Web Documents, IEEE Intelligent Systems, 18(1) (January-February 2003), pp 14-21.

[2] Ben Abdessalem Karaa Wahiba Web-based recruiting (2009). A Framework for CVs Handling. SecondIntemational Conference on Web and Infonnation Technologies "ICWIT'09" June 12-14 2009, kerkennah Island, Sfax, Tunisia pp 395-406.

[3] Berio G, M Harzallah (2005). Knowledge Management for Competence Management. Journal of Universal Knowledge Management, vol. 0, no.1. 2005 pp21-28.

[4] Clech Jérémy, Djamel A. Zighed (2003). Data Mining et analyse des CV : une expérience et des perspectives. EGC 2003 Lyon, France, 22-24 january 2003 pp189-200.

[5] Colucci Simona, Tommaso Di Noia, Eugenio Di Sciascio, Francesco M. Donini, Marina Mongiello, Marco Mottola (2003). A Formal Approach to Ontology-Based Semantic Match of Skills Descriptions. Joumal of Universal Computer Science, vol. 9, no. 12 (2003), pp 1441-1442.

[6] Cunningham, Dr Hamish and Maynard, Dr Diana and Bontcheva, Dr Kalina and Tablan, Mr Valentin (2002). GATE: A Framework and Graphical Development Environment for Robust NLP Tools and Applications. Proceedings of the 40th Anniversary Meeting of the Association for Computational Linguistics (ACL'02), Philadelphia, US, 2002.

[7] Desmontils E., C Jacquin, E Morin (2002). Indexation sémantique de documents sur le Web: application aux ressources humaines Proceedings of Joumees de l'AS-CNRS Web semantique, 2002.

[8] Diana Maynard, V. Tablan, C. Ursu, H. Cunningham, and Y. Wilks (2001). Named Entity Recognition from Diverse Text Types. In Recent Advances in Natural Language Processing 2001 Conference, pages 257 274, Tzigov Chark, Bulgaria, 2001.

[9] Dom J., Naz T. (2007). Meta-search in human resource management. Proceedings of the 4th Intemational Conference on Knowledge System Thailand 105-110.

[10] Feldman R. (2007). The Text Mining Handbook: Advanced Approaches in Analyzing Unstructured Data. Cambridge University Press, 2007.

[11] Kessler Rémy , Nicolas Béchet, Juan-Manuel Torres-Moreno, Mathieu Roche and Marc El-Bèze (2009). Job Offer Management: How Improve the Ranking of Candidates. ISMIS 2009 Prague - 18th Intemational Symposium on Methodologies for Intelligent Systems pp 431-441.

[12] Noia T. Di, E. Di Sciascio, F.M. Donini (2007). Semantic Matchmaking as Non- Monotonic Reasoning: A Description Logic Approach. Joumal of Artificial Intelligence Research (JAIR), 29:pp278–280, 2007.

[13] Trichet, F., Bourse, M., Leclere, M., Morin, E. (2004). Human resource management and semantic Web technologies Information and Communication Technologies: From Theory to Applications. 2004. Proceedings. 2004 Intemational Conference Volume, Issue , 19-23 April 2004 Page(s): 641 – 642.

[14] Yahiaoui L, Z Boufaida and Y Prié (2006). Semantic Annotation of Documents Applied to e-recruitment In Proceedings of SWAP 2006, the 3rd Italian Semantic Web Workshop, Pisa, Italy, December 18-20, 2006 pp1-2.