Structuring Job Search via Local Grammars

Sandra Bsiri[1], Michaela Geierhos[1], and Christoph Ringlstetter[2]

[1] CIS, University of Munich
[2] AICML, University of Alberta

Abstract. The standard approach of job search engines disregards the structural aspect of job announcements in the Web. Bag-of-words indexing leads to a high amount of noise. In this paper we describe a method that uses local grammars to transform unstructured Web pages into structured forms. Evaluation experiments show high efficiency of information access to the generated documents.

1 Introduction

After years of steady growth, the main source of information about job announcements is the Internet [1]. Though, there is some bastion left for high profile openings and local jobs, the traditional newspaper advertisement is of declining importance. Big organizations such as corporations or universities provide an obligatory *career link* on their home pages that leads to their job openings. According to a recent study [2], for example, 70% of the workforce in France searches for jobs on the Internet.[3] The interface arranging access to the information on job opportunities is provided by specialized job search engines. Due to the sheer amount of data, a sophisticated technology which guarantees relevancy would be required. In reality, though, search results are rife with noise.

As compared to a standard search engine, job engines are specialized in that they only index a certain part of the document space: pages that contain job announcements. Unfortunately, in most cases, at this point the specialized treatment of the data has its end. The majority of engines uses variants of the standard vector space model [3] to built up an index that later on is approached by similarity based query processing. This blind statistical model leads often to poor results caused, for example, by homography relations between job descriptors and proper names: in a German language environment a search for a position as a *restaurant chef* (German: *Koch*) easily leads to a contamination with documents that mention a "Mr. Koch" from human resources, with "Koch" being a popular German name. Problems like these arise because the used bag-of-words model treats all terms equally without being aware of their context.

The starting point of a solution is the structured nature of the data spanning the search space and the queries accessing it. Unlike a general search scenario, job search can be seen as a slot-filling process. The indexing task is then to detect concept-value pairs in the HTML-documents and make them accessible. Other

[3] For certain groups such as IT-professionals this value probably comes close to 100%.

Advances in Natural Language Processing and Applications
Research in Computing Science 33, 2008, pp. 201-212

Received 25/10/07
Accepted 07/12/07
Final Version 16/01/08

than fully structured data stored in databases, Web pages for career announcements are set up by humans for humans. Nevertheless, induced by widespread copying of HTML-source-code and corresponding reengineering on account of popular design recommendations, the data are more uniform than is the case for other Web genres: they can be characterized as semi-structured data. Obligatory elements, such as the name of the organization in a home page or the identifier of an announced position combined with a highly restrictive domain vocabulary, make *local grammars* the appropriate tool to discover the logic of a job offer that can then be integrated into a relational structure. As our studies during this research were limited to the Francophone job market, the investigated language was French.

The first step to provide transparent access to announced job openings by an integrated platform was the automatic classification of Web pages into a database of organization home pages. The HTML-structure of these pages was then scanned for anchor elements that lead to job advertisements.

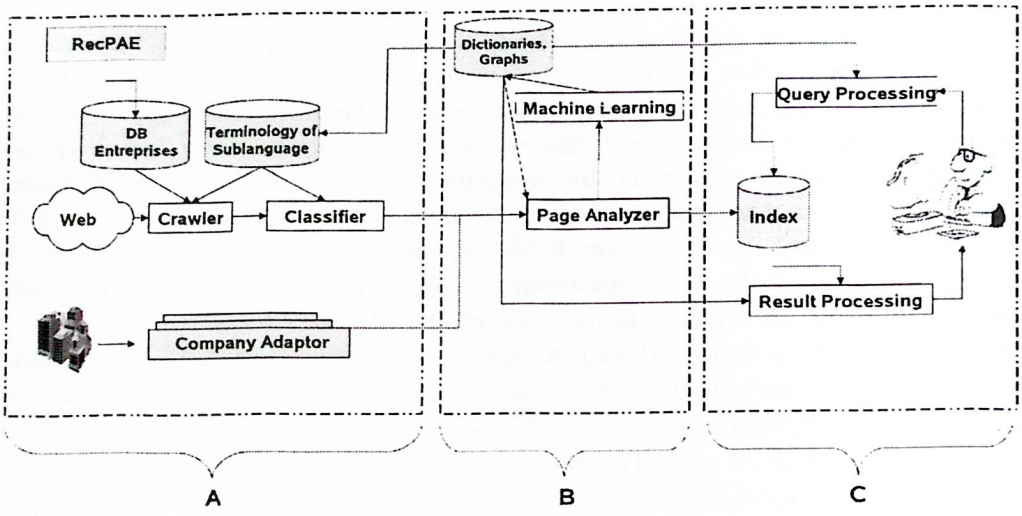

Fig. 1. Overview of the system architecture

Local grammars [4, 5] and electronic lexica [6, 7] were employed for information extraction to transform running text into a semantically structured form. By linguistically motivated analysis, the form slots were automatically filled with values to enable high selectivity for the subsequent retrieval functionality. In Figure 1, we show our architecture for job search that is based on three interactive modules to establish a relational database holding job openings and a query module.

The paper is structured as follows. The next section introduces strategies for the location of job openings on the Web. Section 3 presents methods to extract the structured information of a job announcement and its integration into a frame to enable high selectivity for user queries. In Section 4 we evaluate the

proposed methods. The conclusion comments on practical implications of the given approach and the directions of future work.

2 Locating online job offers

To develop a centralized platform for job search, all relevant job announcement documents on the Internet have to be automatically detected. Two different strategies have been explored: firstly, a system for the recognition of the home pages of organizations was developed. From the home pages, we followed the career links and scanned for job opening pages. Secondly, we used genre-specific terminology, bootstrapped from a development corpus, to run a focused crawler that retrieves and classifies pages to detect job announcements.

2.1 Identification of organization home pages

The first approach to detect published job openings on the Web is based on a database of URLs that has to be updated continually because of the steady dynamics of the job market. An automatic binary classification system was developed that for every input URL decides whether its target falls into the class *organization home page* or not. The classification relies on the following main feature classes:

- Formal HTML-structure of the document
- Evaluation of the URL, meta-information and title
- Analysis and classification of the anchor elements into pre-defined semantic classes (cf. Table 1)
- Identification of the organization name
- Address, phone number, registration number, etc.
- Extraction of typical expressions and complex terminology

Table 1. Selected examples of anchor texts of links in organization home pages.

Carrire *(Careers)*	Nous recrutons *(We hire)* Nos offres d'emploi *(Our openings)*
Produits/Services *(Products/Services)*	Nos Produits *(Our Products)* Accs la boutique *(To the shop)*
Contact *(Contact information)*	Nous contacter *(Contact us)* Pour venir nous voir *(Visit us)* Nos coordonnes *(Contact)*
Socit *(Company Information)*	Notre Socit *(Our organization)* Qui sommes nous? *(Who are We)* Entreprise *(Company)*
Presse *(Press/Media)*	Communiqus de Presse *(Press Information)* La presse et nous *(In Press)* Presse infos *(Press Information)* media *(Media)*
Clients/Partenaires *(Customers/Partners)*	Nos Clients *(Our Customers)* Espace clientles *(Customer Area)* Nos partenaires *(Our Partners)* Relation client *(Customer Relations)*

Phrases and terminology of organizations. In France, every company can be identified by its SIREN-number, a 9-digit code, within the directory of the National Institute for Statistics and Economical Studies (INSEE). Together with the location of the company, the SIRENnumber comprises the identifier for the commerce register. Additionally, the SIRET number, the SIREN identification extended by a code for the class of business and NTVA, the European tax identification number, were used for organization detection. The three formal company identifiers (SIREN/SIRET/NTVA) were extracted with local grammars. Besides that, the system searched for organization-specific standard phrases (*frozen expressions*) that frequently emerge on home pages of organizations. This task is conducted by a multitude of local grammars that allow the modeling of a high degree of syntactic variation. This flexibility could not be reached by standard string search, where a list of collected phrases would be matched with the document, since minor morpho-syntactic variation prevents the match. Bootstrapping with local grammars [8, 9] is a much more promising method, as illustrated by the following example.

- Notre socit , leader mondial sur le march [...]
- Notre socit est leader europen dans le secteur [...]
- Notre socit est leader sur le march mondial [...]
- Notre socit leader dans son domaine [...]
- Notre socit en position de leader au niveau rgional

Though these five phrases are highly different on the morpho-syntactic level, they describe a highly similar context of the word "leader" that refers to the same semantic concept. By a local grammar, as shown in Figure 2, it is possible to efficiently represent all five phrases.

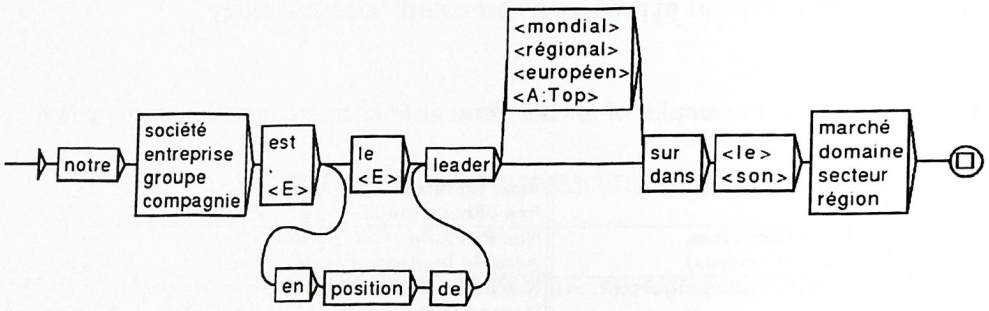

Fig. 2. Local grammar that models the context of the word *leader*.

Organization name and URL. A highly descriptive feature to recognize the home page of an organization is the connection between the name of the organization and the URL of the page. The automatic recognition of company names [10] is a sub-task of *named entity recognition*. Because of ambiguities and sparseness, a list of organization names can not attain sufficient coverage with

regards to the annotation of job announcements. Machine learning systems are extremely dependent on the training corpora or the sub-language used [11, 12].

We used linguistically motivated local grammars to describe organizational contexts and to detect the organization identifier within these contexts. Local grammars [5] enable the description of a local context and restrict the emergence of certain lexical or syntactical features to a window of predefined size. Thus, they avoid or reduce ambiguities that occur for a simple keyword search. To establish a match between the candidates for the organization name and the URL we applied a segmentation algorithm for the domain name. If the result of this segmentation algorithm matches one of the candidates found in the text of the page, it will be regarded as the name of the organization publishing the Web page.

Classification algorithm The retrieved feature value sets are then fed into a classification algorithm that provides a decision on the basis of weighted features as for whether a Web page is an organization homepage or not. When an organization home page has been recognized, it is scanned for links to job openings. This search is guided by the HTML-structure of the page. We introduced a class *careers* which refers to the information provided by anchor texts underlying those links that lead to the openings of the organization. During the learning cycle of the system we found more than 80 different linguistic sequences that lead to a classification into the category *careers*.

2.2 Focused crawler

A second method for the retrieval of job advertisements on the Internet concentrates on the typical terminology of a job offer, which comes close to a sub-language [11, 12]. To this end, during the training phase, frozen expressions and complex phrases were collected. We distinguish two types: nominal phrases that semantically structure the job descriptions and manifest themselves in the headings of the paragraphs of the job announcement, and frozen expressions that contain specific verbs and nouns of the sub-language and that only make sense within the context of job announcements. With this terminological specification, a focused crawler can decide whether a Web page falls into the category job announcement even if the HTML-structure of the document gives no insight whatsoever on the semantic organization of the page.

3 IE and document vectors

The second module of the introduced system concerns information extraction (IE) and the automatic transformation of the textual representation of a job announcement into a semantically structured document. With more than 100 local grammars and electronic dictionaries, an automatic population of a relational database of job announcements is realized. The structure of the database can be

considered as a form that comprises the information contained in a job advertisement. Unlike other systems, the form filling process is fully automatized. From a document that has been recognized as a job announcement by the system (part A), we extract 20 information bits to fill a form that is presented in Table 2.

Table 2. Form of a structured job offer

Example of a job offer form	
Date of publication	22. Jan 2007
Application deadline	fin fvrier *(End of Februay)*
Employment date	mi-mars *(Mid-March)*
Job description	ingnieur d'tude en lectromcanique *(Poject manager (electromechanics))*
Contract type	intrim temps partiel : 1 2 jours/semaine *(Temporary employment:* *1-2 days/week)*
Employment period	8 mois renouvelables *(8 months renewable)*
Workspace	sud-est de Paris *(South-east of Paris)*
Offered salary	selon profil *(Commensurate with experience)*
Job code	MOR34544/ing-21
Professional experience	exprience de 2 3 ans dans un poste similaire *(2-3 years of professional experience* *in a similar position)*
Education	de formation Bac+5 de type cole d'ingnieur *(Engineering degree)*
Company name	CGF Sarl
Office	**Address:** 34 bis rue Berthauds, 93110 Rosny **Phone:** 0 (+33) 1 12 34 45 67 **Fax:** 0 (+33) 1 12 34 45 68 **E-mail:** contact@cgf.fr **Homepage:** http:///www.cgf.fr
Contact	Directeur des RH, Mr. Brice *(HR manager, Mr. Brice)*
Company sector	Construction lectromcanique *(Electro-mechanical construction)*

For the transformation of the initial HTML-documents into the form schema, we need different operations. The chronological application of the five preprocessing steps is shown in Figure 3 (see next page).

Preprocessing. Each document is prelabeled with semantic-structural markers to constrain the application area of the local grammars, such as [TagMISSION], [TagPROFIL], and [TagFORMATION]. During the training phase, 13 semantic classes were established that contain frozen expressions or phrases constituting the surface forms (the textual realizations) of a tag. For example, the label [*TagMISSION*] represents the tasks a position is concerned with. The following expressions can be found:

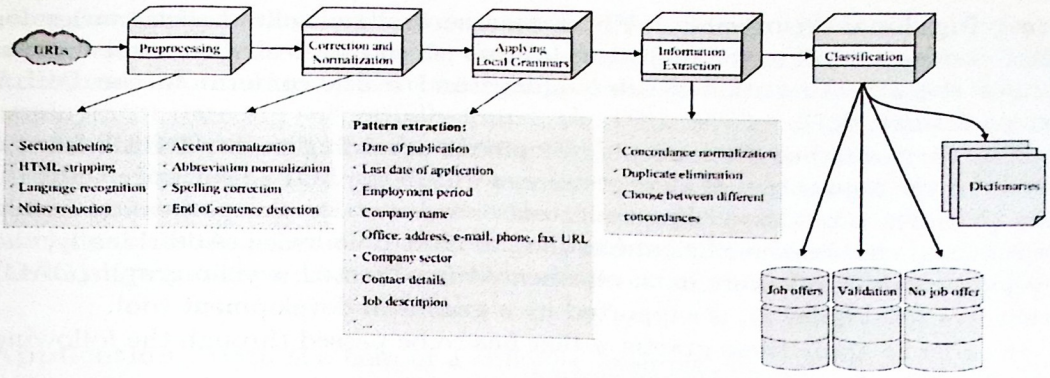

Fig. 3. Steps of the classification and transformation of a possible job annoucement

Your responsibilities
Your duties are
Your field of activities
Duties/activities
We expect
Our expectations
Area of responsibility
We expect from you

During the development of the system we observed three different strategies to compose job advertisements: either frozen expressions are used to structure the document, or we find a mere textual description without a clear structure, and finally a very compact composition with a minimum of information can represent a job advertisement. The third type comprises short announcements such as *welders hired* that do not provide enough linguistic material to be automatically recognized.

After analyzing and labeling the structure, the processed document is stripped of HTML-formating and scripting (JavaScript, ActionScript). Subsequently, the language of the remaining text is classified with the help of the Unitex dictionaries for English, German, and French. If a Web page is composed of less than 50% French words, it is filtered out.

Correction and Normalization. The cleansing step consists of the recognition and correction of orthographic errors with the help of a list of spelling mistakes, which was collected from a large corpus of job advertisements, and the reconstruction of missing accents of French words. During the normalization step, we identify abbreviations and substitute them with their original non-contracted spelling variants, as for example, *ing.* $\mapsto$ *ingnieur, comm.* $\mapsto$ *commercial.* For this purpose, a list of abbreviations and their corresponding non-contracted word sequences was applied. By the end of this processing step, a normalized text is available on which syntactic and lexical analysis can be performed successfully.

Applying local grammars. We created several specialized dictionaries for simple terms as well as for multi-word terms which can deal with a substantial part of the above mentioned sub-language and which conform to the DELA lexicon format [6, 7].

The convention of dictionary development according to the DELA format allows for the application of local grammars within the LGPL[4] software Unitex[5]. This platform provides all linguistic tools necessary for the processing of big corpora and enables the efficient handling of electronic lexica. Additionally, the development of local grammars, represented by directed acyclic graph (DAG) structures (cf. Figure 2), is supported by a graphical development tool.

In order to apply these graphs, a text has to be passed through the following processing pipeline of the Unitex system:

1. *Conversion*: Converts the text into Unicode (UTF-16LE)
2. *Normalization*: Normalizes the special characters, white spaces and line breaks
3. End of sentence detection
4. Dissolving of contractions (e.g. *d'une* $\mapsto$ *de une*)
5. *Tokenization*: Tokenizes the text according to the alphabet of the investigated language
6. *Dictionary*: Performs a lexical analysis by comparing tokens to lexical entries and assigns each word to its possible grammatical categories

The lexical analysis is directly followed by the step of information extraction with local grammars. Each piece of structured information is covered by different extraction units.[6] We created a system of local grammars which work iteratively and sometimes cascaded [13]. The cascaded grammars are executed with different priorities. For example, we developed local grammars to extract job descriptions with eight levels of priority. The graphs of level $n + 1$ are only applied if the grammars of level n with higher priority generate no recall.

Information extraction. The main task of module B (information extraction) is to normalize the extracted sequences and to eliminate duplicates as well as incompletely recognized units. For information extraction, the local grammars organized as DAGs were used. Because sequences are recognized simultaneously by alternative paths of the DAGs that work as annotation transducers a decision on the match strategy had to be made. Since results on training data showed a superior performance, the longest match strategy was used.

Classification. As it is quite unusual that a job offer contains values for all possible information slots, we developed rules modeling the dependency of the

[4] GNU Lesser General Public License

[5] http://www-igm.univ-mlv.fr/unitex/

[6] Among them are, for example, job description, company name, office, workspace, and offered salary.

recognized information in the document for the further classification of job advertisements and the transfer to the database. If several of the semantically-structured information bits are realized by phrases, the job description and the employment date are successfully found, the URL will be classified as a job offer. Additionally, it will be indexed according to the retrieved information in the database. By means of a graphical user interface, the annotated job advertisement can be manually enriched with the missing pieces of information. Several functionalities are provided to interactively expand the semantic dictionaries and the local grammars with new surface realizations of semantic concepts.

Application. With the help of a concrete example, we want to illustrate typical results as well as the quality of the extraction process, driven by the local grammars developed for our system. In Figure 4 we show a tagged document after being passed through all the processing steps described previously.

The recognized surface realizations of semantic classes are labeled by square bracket tags. Our example in Figure 4 shows that all five instances of the semantic classes occurring in the job offer could be located (*TAGCPN = Company information, TAGPOSTE = Job description, TAGEXP = Qualifications of the employee, TAGSALAIRE = Salary, TAGCONTACT = Contact information*).

By means of the more than 100 different local grammars, 14 of the totally 20 sought-after pieces of information (underlined in Figure 4) could be identified and associated to the correct semantic tags:

`<PosteName>` = *Job description*
`<Location>` = *Location*
`<Duree>` = *Contract period*
`<Salaire>` = *Offered salary*
`<CPN>` = *Company name*
`<DomainOrg>` = *Company sector*
`<TypeContrat>` = *Type of contract*
`<Reference>` = *Reference*
`<Contact>` = *Contact information*
`<Prenom>` = *First name of the contact person*
`<NomF>` = *Last name of the contact person*
`<Addresses>` = *Company's address(es)*
`<TEL>` = *Phone number of the contact person*
`<FAX>` = *Fax number of the contact person*

This example shows a sequence not completely recognized by our system which should belong to the concordance of extraction results. Therefore it could not be retrieved as a correct piece of information by our local grammars. According to the automatically gained information, the located place of work would be *"de Paris."* However, it should be *" l'extrieur de Paris"* (the greater area of Paris). During the primary configuration step of our system, the existing semantics has changed and because of that the system's performance suffers a severe setback concerning this special information slot: if a user looks for a position in the city

URGENT ! <PosteName> DÉVELOPPEUR PERL - 94 - FREELANCE </PosteName>(H/F)
FR-IDF-ILE DE FRANCE

Descriptif :
Mon client, un éditeur de logiciel international, recherche de façon urgente un Développeur Perl.

[TAGCPN] La Société :
Mon client est un acteur majeur sur son marché travaillant avec les plus grands comptes internationaux. Suite à une surcharge importante, ils sont actuellement à la recherche d'un développeur Perl qui pourrait démarrer une mission très rapidement.
Mission située à l'extérieur <Location> de Paris </Location> (très facile d'accès par les transports en commun) pour laquelle vous devez impérativement être disponible sous 1 à 4 semaines.

[TAGPOSTE] Description de poste :
Vous devrez tout d'abord analyser plusieurs sites Web ainsi que leurs fichiers attachés puis vous aurez à charge de leur développement sous la derniere version de Perl.
Votre expertise technique, votre implication et votre motivation vous permettront d'évoluer au sein d'une équipe dynamique, pour un client qui apportera une forte valeur ajoutée à votre parcours.
Excellente opportunité de rejoindre une société très demandée, sur une mission de <Duree> 3 mois </Duree> avec de fortes possibilités de renouvellement.

[TAGEXP] Description des Candidats :
- Perl : 2 ans minimum
- Anglais est un plus
- XML : 1 an
- Html : 2 ans

[TAGSALAIRE] Tarif :
<Salaire> 290 à 330€ jour selon expérience <Salaire> .

[TAGCONTACT] Contact :
Si vous avez les compétences nécessaires, merci de me contacter très rapidement afin que je vous organise un entretien avec mon client.

<CPN> Computer Futures Solutions </CPN> est un acteur majeur sur le marché du recrutement et de la prestation de services au niveau Européen dans le domaine des <DomainOrg> technologies de l'information </DomainOrg> avec un chiffre d'affaires de plus de 220 Millions d'euros. Nous sommes présents dans les plus grandes capitales (Paris, Londres, Amsterdam, Bruxelles ...).

Additional Information
Negotiable
Position Type:<TypeContrat> Full Time </TypeContrat>, Temporary Contract Project
<Reference> Ref Code: 391289 </Reference>

[TAGCONTACT] Contact Information
<Contact> <Prenom> Rudy </Prenom> <NomF> Nabet </NomF> </Contact>
<CPN> Computer Futures Solutions </CPN> - Paris
<Addresses> 33 RUE DE LA BOETIE, PARIS 75008 </Addresses>
Ph:<TEL> + 33 1 42 99 83 33 </TEL>
Fax:<FAX> - 33 1 42 99 83 00 </FAX>

Fig. 4. Example of an automatically tagged job offer

of Paris itself, he will be successful because of the produced error. But if the user tended to seek after work outside of Paris, he would have no chance to locate this offer. The same was true for synonymous descriptions of the mentioned phrase. So, the user could have typed the query *"IDF" or "le de France"*, the French

name for the greater area of Paris, which should be mapped to *" l'extrieur de Paris"*, a paraphrase describing the same place. Meanwhile, our system can deal with these linguistic phenomena.

The undiscovered sequences, missed by our local grammars, are gray highlighted. The example shows that the "employment date" paraphrased by two statements like *"vous devez impratiement tre disponible sous 1 4 semaines" (You should be available within 1 or 4 weeks.)* and *"... pourrait dmarrer une mission trs rapidement" (... the occupation could possibly start pretty soon.)* could not at all be located. This lack of information has already been eliminated in a new version of the system and the missing structures were included in the corresponding local grammars. In this manner, the local grammars are continuously expanding which can be realized without great effort thanks to the intuitive Unitex interface [14].

A great advantage of local grammars is the clarity of the already prepared but still missing syntactic structures. If a piece of information was not completely extracted, the missing sub-path can be added quickly to the graphs because of the grammars' high level of modularity.

4 Evaluation

To evaluate the quality of our system with regards to the recognition of information bits indicating job offers, subsequently to the completion of the system, we designed a small, manually annotated test corpus composed of approximately 1,000 job offers.[7] This corpus allows us to provide values for recall and precision of the automatically retrieved search results.

Table 3. Evaluation results gained on test corpora

Extracted type of information	Precision	Recall
Job description	96.9 %	93.3 %
Company name	94.3 %	90.6 %
Office (address)	93.0 %	92.3 %
Salary information	97.1 %	91.8 %
Workspace	98.0 %	96.9 %
On average	95.9 %	93.0 %

Table 3 shows promising results of precision (95.9% on average) and recall (93.0% on average) considering the recognition of entities typically found in job offers. The experimental evaluation presented in this paper was limited to the five highest weighted of the 20 information bits. The majority of difficulties could be observed for the explicit identification of company names.

[7] For research purposes the annotated test corpus is available at `http://www.cis.uni-muenchen.de/~sandrab/DA/IE-Korpus1.html`

5 Conclusion

We presented an integrated platform to enable job search on the Internet. Apart from an overview of the location of job documents on the Web, the main focus of this paper is document analysis and information extraction. The core technique to automatically extract structured information from text is the use of local grammars. The evaluation on an annotated corpus shows excellent results for the proposed approach.

Though the linguistic descriptors and the examples of job advertisements refer to the French job market, the methods are generalizable to other language environments: a system working with equally expressive local grammars will show significant advantages as compared to a mere keyword approach.

References

1. Fondeur, Y., Tuchszirer, C.: Internet et les intermdiaires du march du travail. In: La lettre de l'IRES. Number 67. IRES - Institut de Recherches Economiques et Sociales, Noisy-le-Grand, France (2005)
2. Focus RH: Le guide des 500 meilleurs sites emploi. Jeunes Editions, Levallois-Perret, France (2006)
3. Salton, G., McGill, M.J.: Introduction to Modern Information Retrieval. McGraw-Hill, Tokyo (1983)
4. Gross, M.: Local grammars and their representation by finite automata. In Hoey, M., ed.: Data, Description, Discourse, Papers on the English Language in honour of John McH Sinclair. Harper-Collins, London (1993) 26–38
5. Gross, M.: The Construction of Local Grammars. In Roche, E., Schabs, Y., eds.: Finite-State Language Processing. Language, Speech, and Communication, Cambridge, Mass.: MIT Press (1997) 329–354
6. Courtois, B., Silberztein, M.: Dictionnaires lectroniques du franais. Langues franaise **87** (1990) 11–22
7. Courtois, B., Garrigues, M., Gross, G., Gross, M., Jung, R., Mathieu-Colas, M., Silberztein, M., Vivs, R.: Dictionnaire lectronique des noms composs DELAC : les composants NA et NN. Rapport Technique du LADL **55** (1997)
8. Gross, M.: A bootstrap method for constructing local grammars. In: Contemporary Mathematics: Proceedings of the Symposium, University of Belgrad, Belgrad (1999) 229–250
9. Senellart, J.: Locating noun phrases with finite state transducers. In: Proceedings of the 17th International Conference on Computational Linguistics, Montral (1998) 1212–1219
10. Mallchok, F.: Automatic Recognition of Organization Names in English Business News. PhD thesis, Ludwig-Maximillians-Universitt Munich, Munich, Germany (2004)
11. Harris, Z.S.: Mathematical Structures of Language. Interscience Tracts in Pure and Applied Mathematics **21** (1968) 230238
12. Harris, Z.S.: Language and Information. Bampton Lectures in America **28** (1988) 120128
13. Friburger, N., Maurel, D.: Elaboration d'une cascade de transducteurs pour l'extraction des noms personnes dans les textes. In: TALN 2001, Tours (2001)
14. Paumier, S.: Manuel d'utilisation d'Unitex. (2004)