Information Extraction from Tagged Bibliographical References

Alberto Cáceres Álvarez[1] Alneu de Andrade Lopes[1,2]

[2]ICMC – Instituto de Ciências Matemáticas e de Computação – USP
Av. Trabalhador São Carlense, 400 Caixa Postal 668 – CEP: 13560-970
São Carlos, SP, Brasil
[1]betinho.alvarez@gmail.com [2]alneu@icmc.usp.br

Abstract. We present a rule-based approach for automatic information extraction from bibliographical references in scientific papers. The technique comprises a tagging phase that employs a predefined set of tags to tag all tokens in each reference, followed by extraction of meaningful elements of information (e.g., authors, title, journal, conference, pages, year, etc.) from each tagged reference using a set of rules. The system was evaluated on a corpus with nearly 25,000 references (over 1,000,000 tokens) given in different referencing styles. Performance of the proposed technique on extracting standard fields from references, measured by Precision, Recall and F-measure, is superior to that of state-of-the-art analogous systems reported in the literature.

Keywords: information extraction, POS-Tagging.

1 Introduction

The purpose of Information Extraction techniques (IE) is to identify relevant pieces of information in a document or in a collection of text documents written in natural language. A number of applications, such as field-based search, author analysis, citation analysis, social network analysis, and research community detection require meta-data embedded in paper headers and references, corresponding to individual information slots such as author, title, institution, and so on. Several solutions aimed at automatic information extraction from large collection of papers have been reported in the literature, fostered by current potential applications and the widespread availability of scientific publications on the web. In this scenario, the quality of the information extracted by such systems becomes of great significance. Nevertheless, state-of-art performance of papers' meta-data extraction systems, particularly in the case of extracting information from bibliographical references, is still poor in terms of precision and recall.

In this paper we present an approach for Information Extraction based on the well-known technique of Part-of-Speech Tagging (Brill, 1995; Brill, 1994) that significantly surpasses the performance of known systems in this task. Our extraction approach encompasses a tagging phase, in which all tokens present in a reference is tagged according to a predefined set of 28 tags (e.g., *author*, *title*, *year*, etc.). Following the tagging step, a set of rules written in Practical Extraction and Report Language - Pearl is responsible for deciding whether a token or a sequence of tokens assigned with the same tag (e.g., a token tagged with the tag *year*, or several tokens tagged with *title*) should be extracted as a piece of relevant information. We observe that there is a correlation between the precision of the tagging phase and the precision of the extraction phase, and that the use of a tagger can significantly improve the quality of the IE task.

The remainder of the paper is organized as follows. In Section 2, we review known approaches for information extraction and introduce standard evaluation criteria. In Section 3 we detail the proposed approach, which is based on a generalization of the use of part-of-speech (POS) tagging technique. In Section 4 we present an experimental evaluation and, finally, conclusions are in Section 6.

2 Background on Information Extraction of Bibliographical References

Information extraction (IE) from bibliographical references is a non-trivial task due to variance in the structure of papers in general and of references in particular. A number of techniques has been proposed to deal with this task, which may be categorized into one of two major approaches, rule-based or machine-learning-based techniques.

Day et al. (2005) propose a knowledge-based technique to extract information from references that adopts a hand-made ontology for knowledge representation, named INFOMAP. INFOMAP handles six different reference styles, yielding good results. Nevertheless, it uses only journal papers, and deals with a limited set of seven types of information, namely *author*, *title*, *journal*, *volume*, *number*/*issue*, *year*, and *pages*.

On the machine learning based approaches, an example is the work by Connan and Omlin, who trained *Hidden Markov Models* (HMM) to recognize one of the following reference styles, AAAI, NEWAPA, IEEE. As long as the style is identified correctly, extraction precision reaches 97% (Connan & Omlin, 2000).

Yin et al. (2004) have applied *bigram* HMM to extract information from different reference styles without previous information about the styles. The structure and parameters of the HMM are learned automatically from training examples. Their system achieves a global precision higher than 90%.

Tahasu proposed a stochastic model named the dual variable length output hidden Markov model – DVHMM for feature extraction from references in Japanese obtained using optical character recognition (OCR) software (Takasu, 2003). The model is capable of representing the syntactic structure of references and patterns of OCR errors.

AUTOBIB (Geng & Yang, 2004) is a generic wrapper to extract information from references in the Computer Science field. It employs HMM to obtain structured records from text using HTML (e.g., <a href=...>Pankaj K. Agarwal</a>) to deal with different reference types. Such markup language, however, simplifies treating a critical problem in IE, which is to determinate the beginning and the end of the substring to be extracted (the slot filler). Thus, its precision for extraction by token ranges from 89.10% to 98.90%, depending on the absence or presence of multiple delimiters and HTML tags in the text, respectively. The authors tested the AUTOBIB approach on a small dataset from DBLP, containing 55 records with 1,213 tokens.

Peng & McCallum (2004) applied *Conditional Random Fields* (CRF) to extract information from paper headers. They employed a corpus of 500 references categorized in 13 elements, namely *author*, *title*, *editor*, *booktitle*, *date*, *journal*, *volume*, *tech*, *institution*, *pages*, *location*, *publisher* and *note*. The technique was later improved (Peng & MacCalllum, 2006), but it does not improve information extraction from references.

Barros et al (2009) describe an IE approach based in a two-step classification process for extracting elements from bibliographical references. Firstly, the references are divided into fragments and then the fragments are associated to slots according to terms present in the fragments (initial classification). This classification is refined by a second classifier, now based on a Hidden Markov Model, which try to take into consideration structural relations present in the fragments. This pre-classification-and-refining process for IE is similar to the approach presented here, however we use a part-of-speech tagger for the pre-classification and rules based on regular expressions for refinement. Nevertheless, the best result achieved by Barros and colleagues was a precision rate of 87.48% in a test set with 3000 references while the F-measure (described next) by our approach achieves 98.57% (counting by token), and 96.31% (counting by slot) in a test set with 7500 references.

Evaluating approaches for information extraction from paper references poses additional challenges. The early Message Understanding Conferences (MUC) in the mid-nineties (Sundheim, 1992) defined evaluation metrics for scoring machine and human performance on IE tasks, *Precision* and *Recall* being the most prominent metrics. Precision is defined as the rate between the quantity of information correctly extracted by the total of extracted information, and *Recall* is the rate between the quantity of information correctly extracted by the quantity of relevant information in the text. Another usual metric is *F-measure* (*F-m*), which combines the previous metrics of precision and recall.

The criterion to compute the above metrics may consider information partially extracted. For instance, when the *title* is "*Integrated Case-Based Building Design*" and the slot filler extracted is "*Integrated Case-Based Building*", this may be computed as a correct extraction. We adopt a conservative approach, considering such a return as not correct. We have also adopted the one-slot occurrences – OSO evaluation criterion, since usually slot-fillers in papers have a single value, and we compute extraction countings both for correct tokens and for correct slots extracted. When counting by slot (field-based), the extraction is considered correct only if the slot-filler recovered is complete and correct.

3 Information Extraction from References Based on Induction of Tagging Rules

The task of part-of-speech tagging consists of assigning a tag, from a predefined set of tags, to each token present in a text (word, punctuation mark, equation, etc.), according to the context in which these tokens appear. English words are tagged with their grammatical categories (nouns, verb, etc); punctuation marks are usually tagged with the same symbol (comma, dot, bracket, etc); foreign words, equations, and other features in the text are tagged with a special tag (Eagles, 1996).

We map this process to the problem of extracting information from bibliographical references. The mapping consists of (i) tag all tokens in the references, selecting the appropriate tag from a predefined set, such as *author*, *title*, *journal*, *booktitle*, *address*, *pages*, and *year*; (ii) concatenate sequences of tokens assigned with tags that correspond to slots to be filled; (iii) extract the slot-fillers. Slot corresponding tags in the reference are those with the same name of the slot and/or punctuation marks that bear meaning for the slot-filler being extracted - for instance, the *dot* in author's name abbreviations or the *hyphen* separating the initial and final page information.

The tag set employed includes punctuation marks and other 28 elements, namely *address*, *author*, *booktitle*, *chapter*, *edition*, *editor*, *institution*, *isbn*, *issn*, *journal*, *month*, *note*, *number*, *organization*, *initpage*, *finalpage*, *publisher*, *school*, *series*, *type*, *title*, *url*, *urlaccessdate*, *volume*, *year*, *pages*, *days*, *crossref*.

The experiments were conducted using Eric Brill's TBL tagger (Brill, 1995), a well-known easy to use and free tagger. We notice, however, that the approach is independent of the tagger adopted. TBL is a transformation-based error-driven learning algorithm. It has a training phase with two modules, where the first module induces rules to determine the most likely tag for each token, ignoring its context. The second module induces a set of context sensitive rules, improving the tag assignment accomplished by applying the rules induced in the first module. A possible outcome of the TBL tagger on a reference is shown in the first column of Table 1.

Table 1. Example of tagged and structured reference.

Example of a tagged reference.	Structured reference data in XML format.
Achermann/AUTHOR ,/, F/AUTHOR ./. and/AUTHOR Nierstrasz/AUTHOR ,/, O/AUTHOR ./. (/(2000c/YEAR)/) ./. Explicit/TITLE Namespaces/TITLE ./. In/INDICATOR Gutknecht/EDITOR ,/, J/EDITOR ./. and/EDITOR Weck/EDITOR ,/, W/EDITOR ./. ,/, editors/INDICATOR ,/, Modular/BOOKTITLE Programming/BOOKTITLE Languages/BOOKTITLE ,/, volume/INDICATOR 1897/VOLUME of/SERIES LNCS/SERIES ,/, pages/INDICATOR 77/PAGES -/- 89/PAGES ,/, Zurich/ADDRESS ,/, Switzerland/ADDRESS ./. Springer/PUBLISHER -/- Verlag/PUBLISHER ./. URL/INDICATOR http/URL :/: $b/BARRA $b/BARRA www/URL ./. iam/URL ./. unibe/URL ./. ch/URL $b/BARRA ˜/˜ scg/URL $b/BARRA Archive/URL $b/BARRA Papers/URL $b/BARRA Ache00bExplicitNamespaces/URL ./. pdf/URL	<ref> <author>F. Achermann</author> <author>O. Nierstrasz</author> <year>2000</year> <title>Explicit Namespaces</title> <editor>J. Gutknecht</editor> <editor>W. Weck</editor> <booktitle>Modular Programming Languages</booktitle> <volume>1897</volume> <series>of LNCS</series> <pages>77-89</pages> <address>Zurich, Switzerland</address> <publisher>Springer-Verlag</publisher> <url>http://www.iam.unibe.ch/˜scg/Archive/Papers/Ache00bExplicitNamespaces.pdf</url> </ref>

IE rules in Pearl programming language process a tagged reference combining sequences with the same tags and punctuation marks, and builds each information piece into an XML document as shown in the second column of Table 1.

The example illustrates some peculiarities of the extraction process: punctuation marks, such as comma, between the end of an information and the beginning of another are removed; tokens tagged as *indicator* are also removed (they just indicate presence of an information); when a reference has multiple authors (or editors), these are individually extracted. In this case, specific rules are applied to identify each author or editor. Specific rules, based on the tokens and their tags, are also applied to extract other elements in the references.

4 Results

The proposed approach was evaluated on a corpus with more than one million tokens, including several reference styles such as Plain, Alpha, Abbrv, Apalike, and Chicago, constructed using automatic and semi-automatic procedures, detailed ahead. We notice that there is no benchmark corpus available with the bibliographical references tagged with the adopted tag set.

The corpus has undergone a pre-processing step to handle errors from file format conversion from PDF/PS to TXT[1] and also for data standardization and tokenization. The main tasks were to remove duplicate spaces and tab characters; standardize similar characters (for instance, replacing '__', '--','—', and '_' by '-'); replace the slash character '/' (used by TBL tagger) by $b; and keep just one reference per line, as the conversion process frequently splits a reference into multiple lines. To handle this problem rules are edited to identify the beginning and the end of a reference.

4.1 Training and Test Corpus

The evaluation corpus joins five data sets obtained automatically, and a data set obtained semi-automatically, described in Table 2.

Table 2. Corpus description.

Dataset	Style	# Tokens	# References	Tagging
1	Plain	215726	5000	Automatic
2	Alpha	267679	5000	Automatic
3	Abbrv	219061	4996	Automatic
4	Chicago	220810	4993	Automatic
5	Apalike	177326	3992	Automatic
6	Various	34384	947	Semiautomatic

In constructing datasets 1 to 5 we adopted an automatic tagging process that employs a BibTeX base file containing information from bibliographical references

[1] It has been used the commands *pdftotext* version 3.0 (for linux) and *pstotext* version 1.9 (for windows).

structured into BibTeX fields. Data was retrieved from *The Collection of Computer Science Bibliographies*[2], a collection of scientific literature that encompasses the major topics on computer science.

From this we derived a set of tagged references (training set) with every token assigned with the name of its corresponding field. For instance, each token in the fields (*author*, *title*, *booktitle*, *publisher*, etc.) was assigned with its corresponding field label. Each punctuation mark was assigned with its corresponding mark, and tokens with special meaning for BibTeX were not assigned a tag. For instance, the word 'end' in fields *author* and *editor* is not tagged.

The tagged datasets (1 to 5) were constructed using the assigned BibTeX file, a proper style, and the LaTeX together, deriving a document with the references tagged. Two parameters are necessary to use the BibTeX file: the style (.BST file) and the base (.BIB file). Nowadays, many BibTeX styles are available on the Web. We handle the styles most commonly adopted by the research community available at CTAN[3]. The styles Plain, Alpha, Abbrv e Unsrt are standards. We do not consider the Unsrt style, as it differs from the Plain style only by not sorting the references.

One could suggest that the tagged dataset (1 to 5) could be easier tagged since structured information from BibTeX was used to obtain them. Thus, in order to evaluate the approach also in a set of references from papers on Natural Language Processing retrieved from the Web with random reference styles, we constructed the data set 6 (various). We manually tagged this corpus using a semiautomatic (interactive and iterative) process suggested by Eric Brill in his tagger documentation.

The data set obtained by joining data sets (1 to 6) is referred to as the complete corpus.

4.2 Results on the Complete Corpus

The tagger achieved a global precision of 96.9% on the complete corpus. Due to its size (more than one million tokens), the corpus was split on 70% for training and 30% for testing, with no folding. Tagging results are summarized in Table 3.

Table 3. Tagging results.

Corpus: 24,928 references (1,134,986 tokens) - 70% train 30% test	
Number of tokens in the training set	797,652
Number of tokens in the test set	337,334
Tagging error	10,362
Error rate	3.1%
Precision	96.9%

The corresponding measures of Precision and Recall, and the F-measure in the test set are summarized in Table 4. The average F-measure weighted by frequency achieves 98.57% (counting by token), and 96.31% (counting by slot). We highlight the most relevant information in the table.

[2] http://liinwww.ira.uka.de/bibliography/index.html

[3] http://ctan.org/

[4] http://liinwww.ira.uka.de/bibliography/index.html

Table 4. Extraction results.

	By TOKENS				By SLOTS			
	Freq.	Prec.	Recall	F-m	Freq.	Prec.	Rec.	F-m
	(%)	(%)	(%)	(%)	(%)	(%)	(%)	(%)
ISSN	0.64	99.86	99.99	99.92	2.01	99.70	100.00	99.85
AUTHOR	**12.41**	**99.44**	**99.80**	**99.62**	**13.70**	**99.35**	**99.89**	**99.62**
TITLE	**18.52**	**99.18**	**99.33**	**99.26**	**13.87**	**95.47**	**99.82**	**97.60**
YEAR	**2.20**	**99.07**	**99.28**	**99.17**	**13.82**	**99.74**	**99.52**	**99.63**
PAGES	**3.13**	**98.19**	**99.53**	**98.85**	**11.34**	**94.64**	**99.34**	**96.93**
MONTH	0.80	98.53	99.01	98.77	4.73	98.12	99.54	98.82
EDITOR	1.65	98.97	98.10	98.54	1.93	95.18	98.80	96.96
VOLUME	1.34	98.25	98.51	98.38	7.31	97.62	99.07	98.34
JOURNAL	**4.47**	**98.14**	**97.78**	**97.962**	**5.73**	**93.51**	**99.38**	**96.35**
URL	1.43	96.72	98.60	97.65	0.76	89.01	99.84	94.11
ISBN	0.16	97.57	96.82	97.20	0.26	84.37	100.00	91.52
BOOKTITLE	**7.16**	**96.31**	**97.97**	**97.14**	**6.00**	**87.60**	**99.69**	**93.25**
NUMBER	0.95	96.62	96.86	96.74	5.25	95.86	98.82	97.32
ADDRESS	1.76	94.63	94.78	94.70	5.28	90.74	97.43	93.97
TYPE	0.31	92.71	90.17	91.42	0.86	82.50	95.83	88.67
PUBLISHER	**1.42**	**91.73**	**87.95**	**89.80**	**3.86**	**86.86**	**94.46**	**90.50**
EDITION	0.01	95.54	81.06	87.70	0.06	88.54	84.16	86.29
NOTE	1.17	90.89	81.00	85.66	1.05	56.78	86.08	68.42
SCHOOL	0.18	85.68	79.51	82.48	0.26	54.08	87.13	66.74
INSTITUTION	0.51	78.15	83.81	80.88	0.60	52.06	85.89	64.83
ORGANIZATION	0.54	78.56	83.06	80.75	0.62	53.09	84.14	65.10
INITPAGE	**0.04**	**87.78**	**72.66**	**79.51**	**0.04**	**87.78**	**72.66**	**79.51**
FINALPAGE	**0.04**	**87.44**	**69.10**	**77.20**	**0.04**	**87.44**	**69.10**	**77.20**
SERIES	0.24	82.23	72.48	77.05	0.53	61.29	91.60	73.44
KEY	0.00	94.12	61.54	74.42	0.01	93.75	60.00	73.17
DAYS	0.01	53.54	52.48	53.00	0.08	64.20	90.40	75.08
CROSSREF	0.00	91.89	30.09	45.33	0.02	64.00	42.11	50.79
CHAPTER	0.01	93.85	24.40	38.73	0.05	72.58	62.50	67.16

4.2 POS Tagging and Information Extraction

We analyse how the precision of the tagging process affects precision of the information extraction stage. For this experiment, we employed only dataset 6 (with various reference styles). A 10-fold cross-validation method was applied, since this was the smaller corpus (34,384 tokens).

The global precision achieved was 93.60% in the tagging process. The F-measure for extraction by token was 93.54, and for extraction by slot it was 80.91%.

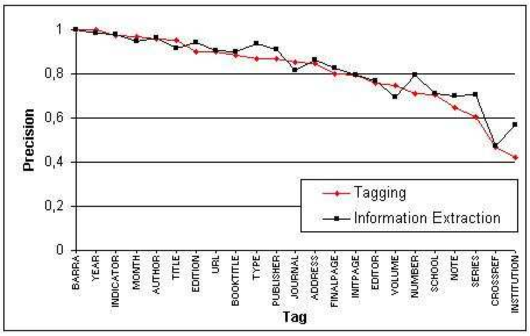

Fig. 1. Precision of the tagging and the extraction processes by tag an by slot.

Figure 1 depicts the precisions of the tagging and of the extraction processes (counting by token) for all relevant slots. One observes a high correlation between precision of the tagging and the extraction processes. Hence, a better tagging precision can improve the quality of the IE process. This result justifies the proposed approach, since POS-tagging is a well-known technique and most current taggers achieve quite good results.

Table 5 summarizes an overall performance comparison between the proposed approach (referred to as POS tagging IE approach) and four recent IE approaches that handle the same task and adopt similar evaluation criteria, namely AUTOBIB (Geng & Yang, 2004), Bigram HMM (Yin et al., 2004), CRF (Peng & McCallum, 2006) e INFOMAP (Day et al., 2005).

Table 5. Overall performance comparison of the proposed IE approach with known IE systems.

	F-measure (by token)	Observation
POS Tagging IE approach	98.57%	References from Various styles, (extraction of 28 fields).
AUTOBIB,	89,10% to 98,90%	Best results depending on the presence of multiple delimiters and HTML tags in the text. (Geng & Yang, 2004)
Bigram HMM,	90.15%	(Yin et al., 2004)
CRF	91.15%	(Peng & McCallum, 2006)
INFOMAP	-	Precision of 97.87%. Only reference from journal, authors do not present the Recall value. (Day et al., 2005).

5 Conclusions

The IE technique proposed significantly surpasses the known approaches described in the literature considering the set of slot-fillers extracted, the variety of reference styles handled, and the average *F-measure* reached.

Unlike similar tagging approaches applied to extract a small number of slot-fillers, the proposed solution generalizes a tagging phase using a well-known technique of POS-tagging to automatically tag all semantic elements in a reference. Excluding punctuation marks, it considers nearly 30 tags, which allows identifying and extracting the most common information pieces in a reference.

We empirically demonstrate the correlation between the precision of the tagging and the extraction processes. Moreover, background knowledge and a pos-processing phase can still improve the results.

References

Barros, F. A., Silva, E. F., Prudêncio R. B, Filho, V. M., Nascimento, A. C. (2009) Combining text classifiers and hidden markov models for information extraction. IJAIT Vol. 18, No. 2, pp 311-329.

Brill, E. (1994). Some advances in transformation-based part of speech tagging. In *Proceedings of the twelfth national conference on Artificial intelligence*, Seattle, Wa.

Brill, E. (1995). Transformation-based error-driven learning and natural language processing: A case study in part of speech tagging. *Computational Linguistics* 21(4), 543–565.

Ding, Y., G. Chowdhury, & S. Foo (1999). Template mining for the extraction of citation from digital documents. In *Proceedings of the Second Asian Digital Library Conference*, Taiwan, 47–62.

Day, M.-Y., T.-H. Tsai, C.-L. Sung, C.-W. Lee, S.-H. Wu, C.-S. Ong, & W.-L. Hsu (2005). A knowledge-based approach to citation extraction. In *Proceedings of the IEEE International Conference on Information Reuse and Integration*, IRI–2005, Vegas, USA, pp. 50–55. IEEE Systems, Man, and Cybernetics Society.

Connan, J. & C. W. Omlin (2000). Bibliography extraction with hidden markov models. Technical report, University of Stellenbosch.

Yin, P., M. Zhang, Z.-H. Deng, & D. Yang (2004). Metadata extraction from bibliographies using bigram HMM. In *7th International Conference on Asian Digital Libraries*, ICADL 2004, Shanghai, China.

Takasu, A. (2003). Bibliographic attribute extraction from erroneous references based on a statistical model. In JCDL'03: *Proceedings of the 3rd ACM/IEEE-CS joint conference on Digital libraries*, Houston, Texas, pp. 49–60. IEEE Computer Society.

Geng, J. & J. Yang (2004). Autobib: Automatic extraction of bibliographic information on the web. In *8th International Database Engineering and Applications Symposium* (IDEAS 2004), pp. 193–204.

Peng, F. & A. McCallum (2004). Accurate information extraction from research papers using conditional random fields. In *Proceedings of Human Language Technology Conference and North American (HLT-NAACL)* pp. 329–336.

Peng, F. & A. McCallum (2006). Information extraction from research paper using conditional random fields. Information Processing and Manegement, 42(2006) 963-979.

EAGLES (1996). EAGLES - Expert Advisory Group on Language Engineering Standards. Recommendations for the Morphosyntactic Annotation of Corpora.

Sundheim, B. (1992). Overview of the fourth message understanding evaluation and conference. In *Proceedings of Fourth Message Understanding Conference (MUC-4).*

Lavelli, A., M. E. Califf, F. Ciravegna, D. Freitag, C. Giuliano, N. Kushmerick, & L. Romano (2004). A critical survey of the methodology for IE evaluation. In *4Th International Conference on Language Resources and Evaluation*.