Towards the generation of a database for scientific research in Natural Language Processing with an Information Extraction system

Iacer Coimbra Alves Cavalcanti Calixto

Dissertation for obtaining the Master Degree in

International Master's in Natural Language Processing and Human Language Technology

Jury

President:	Professor(a) Doutor(a) Full Name
Advisor:	Professor(a) Doutor(a) Henri Madec
Co-advisor:	Professor(a) Doutor(a) Constantin Orasan
Co-advisor:	Professor(a) Doutor(a) Jorge Baptista
Evaluation jury:	Professor(a) Doutor(a) Full Name

20-21 June, 2013

Acknowledgements

I would like to thank, first of all, my supervisors for their aid in this work. I would also like to thank my fellow classmates for their invaluable help – both within class and also outside class, in my daily life. Finally, I would like to thank my family – without them this work could not have been done.

Faro, June 20^{th} 2013

Iacer Coimbra Alves Cavalcanti Calixto

Resumo

A Internet é uma rede de computadores que tem origem no início da década de 1960. Essa rede de redes hoje conhecida como Internet tem lentamente influenciado pessoas e organizações a se interligarem cada vez mais e, dessa forma, possibilitado um crescimento exponencial na quantidade de informação disponível no mundo – uma situação até hoje sem precedentes. Paralelamente ao desenvolvimento da Internet, surge também a área de pesquisa de Processamento de Linguagem Natural (PLN) ou Indústrias da Língua (IL). Na segunda metade do século XX, a área de PLN começou a receber uma grande quantidade de financiamento, sobretudo para condução de esforços no sentido de realizar traduções entre o par de línguas inglês e russo.

O campo de Extração da Informação é um campo da PLN que se presta à realização de extração de informações de forma automática de textos escritos em linguagem natural. Existem vários trabalhos na literatura que utilizam técnicas de Extração da Informação ou Mineração Textual e ao mesmo tempo utilizam-se de publicações científicas como seu corpus principal. Há, assim, alguns trabalhos na literatura que são mais ou menos relacionados ao nosso trabalho – que visa realizar a extração de informações a partir de trabalhos científicos disponibilizados em bases de dados na Internet.

Alguns autores utilizam métodos de Extração da Informação, como Conditional Random Fields, e/ou técnicas de Aprendizagem Automática de Máquina, como Hidden Markov Models, de forma a extrair partes de artigos científicos para variados objetivos. Alguns desses objetivos são busca por campos, análise de autor(es) e análise de citações. Há também aplicações de Support Vector Machines para a extração automática de metadados de artigos científicos.

Há também outros trabalhos na literatura que propõem alguma aplicação de técnicas de Extração da Informação tendo como corpus artigos científicos, mas com um objetivo final claramente distinto em relação ao objetivo do nosso trabalho. [39] propõem um método que usa Hidden Markov Models (HMMs) de forma a extrair informações do cabeçalho de artigos científicos em ciência da computação. [17] propõem um método para a extração automática de metadados de artigos científicos. Eles utilizam Support Vector Machines e, assim como [39], também extraem informações do cabeçalho dos artigos científicos. [4] propõem um método de Extração da Informação para a extração automática de informações da seção de referências de um artigo científico, utilizando como corpus artigos de revistas em farmacologia. [32] propõem um método para extração também de metadados de artigos científicos – assim como título, nomes dos autores e instituições de filiação. Estes autores extraem essa informação tanto do cabeçalho quanto das referências do artigo, ao contrário de [17] e [39], que só utilizam

o cabeçalho. Finalmente, [33] apresenta um método para a extração de nomes de autores de alta precisão. Os autores utilizam tanto as citações textuais como também as entradas bibliográficas na seção de referências do artigo para a extração das informações de interesse.

Nós propomos, neste trabalho, a especificação do sistema IEforNLP (Information Extraction for Natural Language Processing), um sistema de extração de informação. Seu objetivo é extrair informações específicas de bases de dados online de artigos científicos em PLN para a criação de uma base de dados relacional, que deve finalmente ser disponibilizada para uso da comunidade científica. Ainda, nós trazemos detalhes da implementação e da aplicação deste sistema na extração de informações de artigos científicos obtidos no sítio online da Association for Computational Linguistics, assim como uma discussão dos resultados obtidos. Finalmente, nós criamos uma base de dados relacional, populada com as informações extraídas do sítio online da ACL e dos artigos científicos, que será disponibilizada para uso da comunidade.

Nós observamos vários pontos para a definição da lista de publicações a serem utilizadas pelo sistema IEforNLP. Primeiramente, acreditamos que a lista de publicações deva possuir mérito científico. Segundo, que essa lista deva incluir pesquisas sendo realizadas nos melhores centros de pesquisa do mundo, sem distinção geográfica. Finalmente, nós decidimos criar uma lista com a qual pesquisadores concordassem, ao menos teoricamente, como possuidora das características mencionadas. Nós incluimos na lista de publicações somente revistas, simpósios/congressos e workshops sob os auspícios da Association for Computational Linguistics (ACL), de forma a assegurar a qualidade das mesmas.

Nós extraímos as seguintes informações dos artigos científicos obtidos:

(i) Nome e data de publicação do simpósio/congresso ou revista;

(ii) Nomes dos autores;

(iii) Universidade ou instituto de pesquisa de afiliação dos autores;

(iv) País sede da universidade ou instituto de pesquisa;

(v) Campo de pesquisa específico (palavras-chave);

(vi) Informação sobre financiamento.

O sistema IEforNLP é formado por quatro componentes principais: um crawler, um pré-processador, um componente de Extração de Informação (CEI) e um componente de persistência de base de dados. O crawler é responsável pelo download dos artigos de simpósios/congressos, revistas ou workshops. Ele é também responsável pela extração de informações que serão utilizadas pelo CEI posteriormente para a validação da informação extraída por si. O pré-processador é responsável pela conversão dos artigos científicos baixados do sítio da ACL do formato PDF para um formato somente texto, adequado para aplicação do CEI. O CEI, como seu nome já indica, é responsável pela extração de informações dos artigos científicos pré-processados. Esta informação é, por sua vez, repassada ao componente de persistência de dados, que a salvará numa base de dados relacional.

Para a avaliação do trabalho, nós propomos três experimentos distintos:

(i) a avaliação da qualidade do Crawler, parte do sistema IEforNLP, ou seja, o quão bem o crawler identifica e baixa todos e somente os artigos científicos das publicações estabelecidas no experimento.

(ii) a avaliação da qualidade da informação extraída do Componente de Extração de Informação (CEI), parte do sistema IEforNLP, aplicado a artigos científicos da área de PLN.

(iii) a avaliação da qualidade da informação final que é persistida na base de dados relacional. Essa informação é a combinação da informação extraída pelo Crawler e daquela extraída pelo CEI e não é restrita ao texto disponível no corpo do texto. Em outras palavras, isso significa que essa informação inclui todos os campos extraídos pelo Crawler diretamente das páginas do sítio da ACL, assim como dos campos extraídos pelo CEI.

Nós avaliamos a qualidade do componente de extração de informação CEI e a também a qualidade da informação final persistida na base de dados relacional e utilizamos os mesmos dois conjuntos de testes para fazê-lo, um considerado um corpus do mesmo domínio daquele utilizado para a construção do IEforNLP e outro um corpus de um domínio distinto. Pode-se ver pelos resultados da avaliação que, em primeiro lugar, o Crawler funciona exatamente como o esperado; segundo, que mesmo que o CEI não tenha obtido resultados ótimos, atinge resultados globais razoáveis, como indicado pelo F-measure de 84.4%; finalmente, que a qualidade da informação final salva na base de dados relacional é muito boa, como medido pelo F-measure global de 93.8%. Ainda, como esperado, os resultados obtidos com os experimentos no corpus do mesmo domínio daquele utilizado para a construção do IEforNLP mostraram-se melhores que aqueles obtidos no corpus de domínio distinto.

A informação salva na base de dados possui um F-measure de 100% tanto para o nome da publicação quanto para a data, que são informações extraídas pelo Crawler diretamente do sítio online da ACL. Ainda, os nomes de autores também são extraídos pelo Crawler, e os bons resultados obtidos com a sua extração é consequência direta do quão bem o Crawler extrai informações do sítio online da ACL.

Acredita-se que proporcionar à comunidade recursos como a base de dados relacional com informações de pesquisa, como fazemos neste trabalho, é de grande valor para futuras pesquisas relacionadas. Espera-se que essa informação histórica seja útil no auxílio na identificação de oportunidades, para pesquisadores e instituições de pesquisa, e também na identificação de tendências de pesquisa. Esta é, na verdade, uma das razões principais para a criação do IEforNLP. Acredita-se que essas informações sejam desejáveis não só porque elas possam ajudar na identificação de potencialidades e possíveis oportunidades para a comunidade, mas também porque ela provê dados quantitativos com os quais instituições financiadoras de pesquisa e outros interessados possam melhor planejar e melhorar as pesquisas em PLN.

Abstract

THE Internet is a network of computers which dates its origins back to the 1960s. This network of networks known as the Internet slowly began to drive organisations and people towards interconnectedness and set the foundations of a constant growth in the quantity of information available – an unprecedented situation until then. In parallel with this development the field of Natural Language Processing (NLP) was receiving a fair amount of funding, specially for efforts of translation for the Russian-English language pair.

Information Extraction (IE) is an NLP field concerned with the automatic extraction of information from text written in natural language. There are several different works that use Information Extraction or Text Mining techniques and use scientific papers as the main corpora they work upon. There are some proposals in the literature which are more or less related to our task, that is, the task of extracting information from scientific papers in Natural Language Processing publications.

Some authors use Information Extraction, such as Conditional Random Fields, and/or Machine Learning techniques, such as Hidden Markov Models, to extract portions of papers important to tasks such as field-based search, author analysis and citation analysis [32, 39]. There are also applications of Support Vector Machines for automatically extracting metadata from research papers [17]. There are other works available in the literature which also propose some kind of application of Information Extraction techniques on a corpus of scientific papers, but with a final objective clearly distinct from ours.

[39] propose a method that uses Hidden Markov Models (HMMs) in order to extract information from headers of computer science research papers. [17] propose a method for the automatic extraction of metadata from research papers. They use Support Vector Machines (SVMs) and, like [39], also extract information from the paper header. [4] present an Information Extraction method for the automatic extraction of information from the references of scientific papers from journals in pharmacology. [32] propose an Information Extraction system to extract what they call research papers' metadata – such as title, author, and institution. They extract that information from both the paper header and its references – as opposed to [17] and [39], which only use the paper header. [33] present a method for high accuracy citation and author name identification. They use both in-text citations and the references section of the papers in order to extract the bibliographic information they are interested at.

We shall present in this work the specification of the system IEforNLP, an Information Extraction system that aims at the extraction of specific information from online databases of scientific papers in Natural Language Processing. Additionally, we provide details on its implementation and its applica-

tion on the extraction of information from online databases of selected NLP journals, conferences and workshops, along with a discussion of the obtained results. Finally, we discuss the creation of a database populated with the information extracted, which shall be made available for further use.

We observed several points to define the list of the publications used in IEforNLP. We believe, first, that the list of publications to be analysed must be scientifically sound. We also believe the list should encompass research being conducted in all the best research centres in the world. Finally, we aim at creating a list which researchers should (at least theoretically) agree upon as both scientifically sound and comprehensive. We included only journals, conferences and workshops under the ACL umbrella in order to assure the quality of the papers.

We extract the following information from the textual representation of the papers:

(i) Conference or journal name and date of publishing;

(ii) Authors' names;

(iii) University or research institute of affiliation for each of the authors;

(iv) Country of the university of research institute;

(v) Specific field of research (keywords);

(vi) Funding information.

The system IEforNLP consists of four main components: a crawler, a pre-processor, an Information Extraction component and a database persistance component. The crawler is responsible for the download of papers from selected conferences, workshops and journals. It is also responsible for extracting information that will be used to validate extractions made by the Information Extraction component. The pre-processor is responsible for converting the downloaded papers [1] into a text-only format suitable for the Information Extraction component. The IE component, as its name already implies, is responsible for the extraction of information from the pre-processed research papers. This information is then handed in to the database persistance component, which saves everything into a relational database.

In order to evaluate our work, we conducted three different experiments:

(i) we evaluate the quality of the Crawler, that is how well does the crawler download only and all the correct papers for the chosen NLP scientific publications.

(ii) we evaluate the quality of the Information Extraction Component (IEC) of the system IEforNLP when applied to NLP scientific research papers.

(iii) we evaluate the quality of the final information saved into our relational database. This information is the combination of the information obtained by the Crawler and the IEC and is not restricted to the texts of the papers. In other words, it includes all the fields extracted by the Crawler from the ACL webpages as long as the fields extracted by the IEC.

[1] The papers downloaded are available under the PDF format.

We evaluate the IEC and the final information saved into the database using the same two test sets, one corpus to assess how the system performs on an *in-domain* corpus and how it performs on an *out-of-domain* corpus. We can see from the results obtained on the evaluation that, first, the crawler works exactly as expected; second, that even though the IEC does not perform so well, it achieves reasonable overall results, as indicated by the F-measure of 84.4%; finally, that the final quality of the information saved into the database is very good, as measured by the overall F-measure of 93.8%. Additionally, as expected, the results obtained with the experiments on the *in-domain* corpus showed better results then the ones on the *out-of-domain* corpus.

The information that is actually saved into the database has 100% F-measure for publication name and year, which were part of the information obtained by the crawler directly from the ACL website. Also, the authors' names are extracted by the crawler, and their high final results are a consequence of the fact the crawler performs really well in extracting these information from the ACL website.

By specifying our corpus in a restrictive way, we could obtain statistical measures comparable to the ones discussed in the state-of-the-art. The results obtained both with the *in-domain corpus* and the *out-of-domain corpus* show both the qualities and flaws of the system proposed, as can be seen by the reasonable values obtained for the statistical measures computed. The quality of the overall information saved into our relational database evidences the suitability for the extracted information for being used afterwards by decision-makers and/or the research community.

We believe that providing the community with a database of researchers and publications, as we are doing in this work, is highly desirable for future related research. We expect that the historical information that have been put available to the community be useful to identifying opportunities for researchers and institutions and also for foretelling trends in research.

Finally, we believe the database created with the IEforNLP system will be a valuable tool for researchers world-wide. This is actually one of the main reasons we decided creating IEforNLP for, in the first place. We believe it is highly desirable to have detailed historical information on the research being conducted not only because it might help in the identification of strenghts and possible opportunities for the community, but also because it might provide the (quantitative) grounds for funding bodies and stakeholders to plan and improve research in NLP.

Palavras-Chave

Keywords

Palavras-Chave

extração da informação
mineração textual
análise textual
processamento de linguagem natural
linguística computacional

Keywords

information extraction
text mining
text analytics
natural language processing
computational linguistics

Table of Contents

List of Figures

List of Tables

List of Acronyms

Acronym	Designation in English	Designation in Portuguese
ACL	**A**ssociation for **C**omputational **L**inguistics	
ALPAC	**A**utomatic **L**anguage **P**rocessing **A**dvisory **C**ommittee	
ARPA	**A**dvanced **R**esearch **P**rojects **A**gency	
ARPANET	**A**dvanced **R**esearch **P**rojects **A**gency **N**etwork	
DARPA	**D**efense **A**dvanced **R**esearch **P**rojects **A**gency	
DBMS	**D**atab**a**se **M**anagement **S**ystem	Sistema Gerenciador de Banco de Dados
DFM	**D**imensional **F**act **M**odel	
HMM(s)	**H**idden **M**arkov **M**odel(s)	Modelo(s) oculto(s) de Markov
HTTP	**H**yper**t**ext **T**ransfer/**T**ransport **P**rotocol	
IE	**I**nformation **E**xtraction	Extração de Informação
IP	**I**nternet **P**rotocol	
ME	**M**aximal **E**ntropy	
MUC	**M**essage **U**nderstanding **C**onference	
NER	**N**amed **E**ntity **R**ecognition	
NIST	**N**ational **I**nstitute of **S**tandards and **T**echnology	
SCFG	**S**tochastic **C**ontext-**F**ree **G**rammars	
SQL	**S**tructured **Q**uery **L**anguage	
TCP	**T**ransmission **C**ontrol **P**rotocol	
TCP/IP	**T**ransmission **C**ontrol **P**rotocol/**I**nternet **P**rotocol	
TM	**T**ext **M**ining	
TREC	**T**ext **R**etrieval **E**valuation **C**onferences	

MT	**M**achine **T**ranslation	Tradução automática
NLP	**N**atural **L**anguage **P**rocessing	Processamento de Língua Natural

List of Terms

Term	Meaning
Corpus	A collection of written or spoken utterances in machine-readable form, assembled for the purpose of automatic processing.
F-measure	An evaluation measure that combines *Precision* and *Recall*.
Precision	An evaluation measure of correctness. It measures the proportion of correct answers provided by a system over the set of all answers *found* by that same system.
Recall	An evaluation measure of completeness. It considers the proportion of correct answers provided by a system over the set of "true" correct answers.

Chapter 1

Introduction

1.1 Context

THE Internet is a network of computers which dates its origins back to the 1960s. In the U.S. the ARPANET was a network created to connect universities' research centres, which rapidly grew to incorporate more and more centres and universities within the years. It was the result of research being conducted by the United States Defense Advanced Research Projects Agency (DARPA) [1], at the time known as ARPA.

In 1974, the TCP/IP protocol [2] – whose name derives from two of its most important protocols, the Transmission Control Protocol (TCP) and the Internet Protocol (IP) – was publicly released and provided a common infrastructure to connect computers and networks. In the 1980s, the ARPANET fully adopted the protocol specification, thus evolving and allowing for incorporating more nodes. This development pushed the ARPANET towards becoming the widespread, ubiquitous network of networks nowadays known as the Internet.

This network of networks known as the Internet slowly began to drive organisations and people towards interconnectedness and set the foundations of a constant growth in the quantity of information available – an unprecedented situation until then. In parallel with this development the field of Natural Language Processing (NLP) was receiving a fair amount of funding, specially for efforts of translation for the Russian-English language pair.

One of the most well-known pessimistic prospects faced by the NLP community was the release of the Automatic Language Processing Advisory Committee (ALPAC) report in 1966 [3]. Back at that time, a large part of the publicly funded research in NLP in the U.S. focused on the translation of texts from Russian to English. In this report, the committee analyses the results of the most prominent research in NLP conducted in the U.S., concluding in a very discouraging tone.

Hutchkins (1996) [4] emphasises that the report concluded, amongst other things, that there has not been any real machine translation *per se*, when analysing general scientific text; also, that there is no

[1] http://www.darpa.mil/ Last access on March 9, 2013.
[2] http://tools.ietf.org/html/rfc675 Last access on March 9, 2013.
[3] http://www.nap.edu/openbook.php?isbn=ARC000005 Last access on March 9, 2013.
[4] http://www.hutchinsweb.me.uk/MTNI-14-1996.pdf Last access on March 9, 2013.

prospect at all that such a system is bound to appear in the near future; and finally that early translations of selected texts were deceptively encouraging because they allured qualities that did not exist, such as generality and uniformity. The practical results of the publication of the report was a strong decrease in the amount of funding for NLP research in the U.S. in the next decades, and also a growth in disbelief at MT by the public and the academia.

Natural Language Processing (NLP) is a domain of study which proposes the automatic (and semi-automatic) manipulation of natural language with the aid of computers. Therefore, it can be seen as an area of intersection between Linguistics and Computer Science.

The NLP research domain has subsequently originated several different branches in more or less specialised research fields. The ones analysed in this work are the research areas of Machine Translation (MT), Information Extraction (IE) and Text Mining (TM).

1.2 Objectives

The objectives of this thesis are:

- The presentation of a literature review on the field of Information Extraction.
- The specification of the system IEforNLP, an Information Extraction system that aims at the extraction of specific information from online databases of scientific papers in Natural Language Processing.
- The implementation of IEforNLP and its application on the extraction of information from online databases of selected NLP journals, conferences and workshops, along with a discussion of the obtained results.
- The creation of a database populated with the information extracted.

1.2.1 IEforNLP

As previously mentioned, one of the objectives of this thesis is the specification and implementation of the system **IEforNLP** (**I**nformation **E**xtraction **for N**atural **L**anguage **P**rocessing). IEforNLP is an Information Extraction system applied to the problem of extracting information from online databases of scientific papers published in selected Natural Language Processing journals, conferences and workshops. It can be divided in two parts:

(i) the automatic download of research papers in NLP and

(ii) the extraction of some information from these papers.

The abovementioned research papers are downloaded directly from the Association for Computational Linguistics (ACL) website [5], which makes them available for selected conferences and journals under the ACL umbrella. The ACL is probably the most important community of researchers in Natural

[5] http://www.aclweb.org/

Language Processing worldwide, organising one journal, as well as several conferences and workshops. The papers published in journals, conferences and workshops under the ACL umbrella are available for free in the ACL website under the Portable Document Format (PDF) format.

The definition of the specific journals, conferences and workshops included for the extraction task took into consideration several variables. Some of them are:

- whether their papers are available for download free of charge;
- for how long did the publication exist;
- how well is the publication perceived by researchers;
- amongst others.

The reasons we choose specific publications and not others, amongst other architectural and systemic choices, are discussed further in Chapter 3. After automatically downloading papers available in ACL publications from the ACL website, the IEforNLP system automatically converts the PDF files into a text-only format.

We extract the following information from the textual representation of the papers:

(i) Conference or journal name and date of publishing;

(ii) Authors' names;

(iii) University or research institute of affiliation for each of the authors;

(iv) Country of the university of research institute;

(v) Specific field of research (keywords);

(vi) Funding information.

It is important to remember that most of the papers have funding information *(vi)* available, and this information is usually an acknowledgement of the funding institution and a project code under the same institution. We automatically extract the name of the funding institution and the project code in the funding institution the publication refers to. All the information numbered *(i)-(vi)* is automatically extracted and saved into a relational database. We thoroughly discuss these choices and implementation details in Chapter 3.

In Chapter 4 we compute some statistical measures upon the information extracted with IEforNLP and discuss the results obtained and aspects of the research being conducted in NLP.

NLP has been a field that has attracted the attention of governments and public funding agencies ([14, 23, 30]) all over the world since its early days. The possible outcomes of some of the main NLP subfields – such as Machine Translation, Information Extraction or Text Mining –, constitute strategic technologies for many countries. Therefore, the fact that DARPA, the European Community and other major funding bodies have been playing an influential role in the NLP research is not a coincidence.

Bearing that in mind, we want to build a database with the data extracted from the online databases research papers. We believe this database is interesting for researchers who want to study research mobility, productivity and/or trends in the field. In Chapter 5, we discuss how this database shall be used, in the future, to answer questions such as whether there are any trends in the directions researches are taking in NLP.

1.3 Thesis Structure

The remainder of this document is structured as follows:

- In Chapter 2 we present a literature review on Information Extraction and related fields of research.
- In Chapter 3 we introduce the architecture of the IEforNLP system and contextualise it in the best practices described in the literature. We also present and discuss the corpus adopted for this work.
- In Chapter 4 we present an evaluation of the system and a discussion of some statistical measures computed upon the information extracted by IEforNLP. Computational experience is also discussed.
- Finally, a summary and some conclusions complete this thesis in Chapter 5. We also present some ideas for future work.

Chapter 2

State of the Art

2.1 Context

We shall first introduce some of the works on the broad fields of Information Extraction, Information Retrieval and Text Mining. Since part of the objectives of this work is the definition and implementation of an Information Extraction system, the main works on Information Extraction are reviewed. Additionally, the related fields of Information Retrieval and Text Mining are also defined and broadly reviewed.

2.2 Information Extraction

2.2.1 Overview

Information Extraction (IE) is a field concerned with the automatic extraction of information from text written in natural language. According to [11], the engine of an Information Extraction system is implemented as a statistical model, a rule model or a hybrid.

Some examples of the information one might want to extract are entities, relationships between entities and attributes of those entities. Roughly, entities are the concepts one is looking for (i.e., a person, organisation, object). Examples of possible entities are John (person), apple (object) or Google (organisation). Relationships are relations between entities and can be given by a verb (i.e., "John" eats an "apple"). Finally, the so called entity attributes are information that categorise or specify our entities. One example of attributes are semantic information about them, such that the example "John eats an apple" might be enriched with the information that the entity represented by "John" must be a person and the one represented by apple must be something edible ([38]).

Following the explanation on entities, relationships and attributes, [11] states that there are four elements which can currently be extracted from text:

- Entities are the most basic elements one can find in a text. These can be the names of people, organisations, locations, etc.

- Attributes are information describing some of the entities. Examples: "John is big", or "IBM has 30,000 employees", or "Brazil has 200 million inhabitants".

- Facts are truth statements that links two or more entities. Some examples are "John works for IBM", or "Baden Powell was born in Brazil".

- Events are activities that are of interest for the text analyser and will usually involve one or more entities. Examples can be someone's birthday, the holding of a conference at a specific date, etc.

For more detailed definitions for entity, relationship, attribute and other important concepts, refer to the ACE website [1], [1], [7] and [19].

2.2.2 Architecture and Evaluation

Following the discussion of [11] and [29] apud [11], there are problems researchers commonly face when trying to evaluate Information Extraction systems, such as: the reproductibility of the results (what is the exact split of the training/test set), what constitutes an exact match (how to treat a missing comma, for example) or which features to select.

According to [11], the typical architecture of an IE system would consist of three or four major components. These components are applied in a pipeline, one after the other, so that the output of the n-th component is the input of the $(n+1)$-th component. We shall use the architecture devised by [11] in order to discuss the so-called "typical Information Extraction system" architecture. In the next paragraphs, we shall discuss and draw upon the definitions of [11].

The first module of an IE system is the Tokenization component. It is responsible for receiving raw text as input and for outputting a splitted representation of this text according to its "building blocks". These "building blocks" are usually words, but they might also be sentences, paragraphs or whatever one thinks is the most appropriate for the task.

The second module in the pipeline is the Morphological and Lexical Analysis component. This component performs two main tasks, part-of-speech (POS) tagging and sense disambiguation. The part-of-speech tagger will assign a category or part-of-speech to each word on the text. The sense disambiguation will identify all ambiguous words, or all words that can receive more than one part-of-speech tag, and choose the correct POS tag for the text being analysed. Some examples of ambiguous words are *can* (which can be either a *noun* or a *verb*) or *fool* (which can also be either a *noun* or a *verb*).

The third module is the Syntactic Analysis component, responsible for the tasks of Shallow and Deep Parsing. The task of the Shallow Parser is to identify the noun groups and verb groups. The task of the Deep Parser is, given the noun groups and verb groups, find the correct relationships between them.

The noun groups are any sequence of nouns or noun phrases that one might wish to put together. In order to illustrate this, consider the example given by [11, p. 107]. Say we wish to put the appearances of "Company, Location" together in one noun group:

[1] http://www.itl.nist.gov/iad/894.01/tests/ace/ Last access on March 11

> Associated Builders and Contractors (ABC) today announced that Bob Piper, co-owner and vice president of corporate operations, Piper Electric Co., Inc., Arvada, Colo., has been named vice president of workforce development.

The two Noun Phrases (NP) "Piper Electric Co., Inc." and "Arvada, Colo." match the pattern "Company, Location" we search for. We could then make them one noun group. The same principles are applicable to verb groups.

After the identification of the noun and verb groups, the Deep Parser is applied. As already mentioned previously, the Deep Parser generates relationships between the elements of the sentence (namely, the noun and verb groups). These relationships are generally built using domain-specific information and can consist of regular expressions.

Finally, a last module in the Information Extraction pipeline consists of a Domain Analysis component. The Domain Analysis component performs the tasks of Anaphora Resolution and Integration.

The problem of Anaphora or Co-Reference Resolution is the one of identifying whether distinct mentions of an entity in a text refer to the same concept (or object, person, organisation, idea) in the real world. See, for example, the example below:

> George and Alice were studying a lot in the past few weeks. Their mother had promised them that they would go on a nice summer trip if they all had good grades.

In this example, "George and Alice" are a noun group. The mention "their", in the second sentence of the example, refers to "George and Alice" and we call "George and Alice" and "their" co-referential. The mention "them", in the second sentence, is also co-referential with "George and Alice" and therefore with the mention "their". The other two mentions "they", both in the second sentence of the example, are also co-referential with "George and Alice", "their" and "them".

Anaphora Resolution is a difficult problem and, according to [11], is a key element in the pipeline of advanced text mining systems.

2.2.3 Methods and Systems

There are algorithms for IE that attempt at inducing rules given an annotated corpus [11]. One of these systems was proposed by [42] and is called WHISK, and it is discussed ahead.

According to [11], Structural IE is a modality of Information Extraction system that focuses on trying to extract information from a text based on the visual layout of the document that contains that text. They also emphasise that this approach does not aim at substituting other more in-depth IE methods, but rather complement the more conventional text mining techniques.

[11] states that the most prominent probabilistic models for Information Extraction are Hidden Markov Models (HMM), stochastic context-free grammars (SCFG) and maximal entropy (ME) models.

[36], and later [11], defines a Hidden Markov Model as a special type of finite-state automaton, namely one which possesses stochastic state transitions and symbol emissions. In more practical terms, a HMM will begin in an initial state, emit a symbol selected by that state, make a transition to a new

state, emit a symbol selected by that state, make a transition to a new state, and so on, until it reaches a final state. Hidden Markov Models require the usage of probability theory and statistics and showing its internal processes is not in the scope of this work. Example of a problem for which HMMs have been applied are field extraction ([14, 15] apud [11]). According to [30], one of the widespread uses of HMMs is in tagging tasks (such as in part-of-speech tagging).

According to [11], Stochastic Context-Free Grammars can be defined as a normal context-free grammar with the addition of a probability function P to each of its production rules. Grammars can be (and often are) ambiguous, so a certain sentence can be generated by one same grammar using different production rules. When using SCFGs, since we have probabilities assigned for each production rule, we can calculate the different parsing probabilities for all the possible different trees for a given sentence. It is not possible to do that with ordinary context-free grammars.

[11] mention other models applicable for the problem of Information Extraction: Maximal Entropy (ME) Models, Maximal Entropy Markov Models (MEMM) and Conditional Random Fields (CRF). ME models attempt at modeling a random process. There is contextual information available for that process, and we want to know what will be the value of a certain variable output. In order to solve ME models one must solve a constrained optimisation problem. A MEMM is a Hidden Markov Model that, instead of having separate transition and emission probabilities, has only transition probabilities. Additionally, these transition probabilities are dependant on observations. Conditional Random Fields are a type of model based on maximal entropy, like MEMMs, but allow for a better trade off at different sequence positions. We do not discuss these models in greater details any further for they require a extensive knowledge of probability theory and statistics and are therefore out of the scope of this work.

2.2.4 Information Extraction and Information Retrieval

Information Extraction is concerned with the extraction of (structured) information from a set of documents written in natural language (unstructured text). Information Retrieval (IR) is the task of retrieving documents, usually written in natural language (or some markup language in the case of the World Wide Web), given a user query. Thus, it is important to distinguish between both.

Information Retrieval systems can be sub-categorised in more specific classes, depending on the needs one aims to achieve. In earlier IR systems the user query was written by concatenating keywords in a complex query using the boolean connectors *and*, *or*, *not*. Nowadays, IR systems usually allow for a more natural user query (i.e., by allowing the usage of a query written in a quasi-natural language) and perform a *ranked retrieval*. Such retrieval returns a ranked set of documents, which should be identified according to its usefulness for a query ([40]). Whereas IE systems aims at extracting information from a set of documents, IR systems aim at retrieving documents (from an even larger set of documents) given a user query.

Historically, some academic events strongly influenced the fields of Information Extraction and Information Retrieval. According to [34], the U.S. agencies involved in funding research in NLP regarded evaluation efforts as the best path towards high quality standards and scientific recognition. The evaluations set a common ground for participating systems and usually provided them with a corpus for

training, a clear description of the task and how they were to be quantitatively compared.

According to [40], the field of Information Retrieval underwent important developments in the 1960s, when models that allowed for good system responsiveness appeared; in the 1970s and 1980s, new models and techniques were proposed and successfully tested on collections of around several thousand articles – which are nowadays considered a small set for the task; in the 1990s, the movement towards applied technologies received a strong impetus with the inception of the TREC conference in 1992. The Text Retrieval Evaluation Conferences (TREC), a series of conferences with the goal of evaluating retrieval systems, had a positive influence on research in the field of Information Retrieval. The TREC conferences were sponsored by U.S. government agencies under the National Institute of Standards and Technology (NIST) and has had the goal of encouraging research in IR with large-scale text collections ([40]).

[45] argues that by the end of the twentieth century, the field of Natural Language Processing as a whole passed by a paradigm change: it was gradually becoming less theoretical (or less motivated by linguistic theory) and more motivated by real-world applications (or by applied technology). A greater deal of the forces pushing research in NLP towards that new trend are, according to the same author, the TIPSTER, TREC and MUC programs, in that case more closely in Information Extraction and Information Retrieval. [45] presents a literature review on the Information Extraction systems at the time their article was published and we will discuss the most prominent of them to present a panorama of the field from the late 1980s until the early 1990s.

SCISOR, or "System for Conceptual Information, Summarization, Organization and Retrieval", combines methods derived from Artificial Inteligence (AI) and also other techniques, and participated in the MUC-2 conference, even though with no quantitative results available. Its authors present good numbers for domain-specific information extraction ([24] apud [45]).

FASTUS is a system which had years of development as a research project and was heavily based on finite state technology ([22] apud [45]). Moreover, it was a success during its time and influenced several other systems afterwards; it participated of the MUC-6 conferences and achieved results of $F = 0.94$ (Named Entity Recognition task), $F = 0.75$ (Template Entity task), $F = 0.65$ (Co-reference task) and $F = 0.51$ (Scenario Template task) ([45]).

Finally, the University of Massachusetts also presented an important system ([12] apud [45]) at the time, in the MUC-6 conference. It was composed of several modules that accomplished, each, different tasks of the NLP pipeline. Some of these modules include a module for Named Entity Recognition (NER), a part-of-speech tagger and a sentence analyser, amongst others. They also provided scores for the different tasks of the MUC-6 conference: $F = 0.85$ for Named Entity Recognition, $F = 0.61$ for Template Entity, $F = 0.46$ for Co-reference resolution and $F = 0.40$ for Scenario Template ([12] apud [45]).

Other systems for performing information extraction from speech were mentioned in [45], but since we do not attempt at analysis speech in this work we do not discuss it here [2].

Some of the aforementioned academic events that influenced research in Information Extraction are

[2]For more details, refer to [45].

the Message Understanding Conference (MUC) ([6, 19, 43]) and the Automatic Content Extraction (ACE) conferences ([1, 10]). The MUC conferences were a series of conferences that aimed at bringing researchers together in a common effort to increase the efficiency of IE systems. The first conference of the series, MUC-1, was held in 1987 and the last one, MUC-7, was held in 1997. The ACE and the MUC conferences, in general, have similar objectives and addresses the same issues ([1]). The ACE conferences can be seen as a continuation to the MUC effort. The first discussions on the ACE conference started in 1999, and the last conference was held in 2008.

The quantitative indicators adopted to measure how well a system performs are usually precision, recall and F-measure. *Precision* computes to what extent the system identifies only correct instances amidst the whole set of possible instances (i.e., precision is a measure of correctness). *Recall* computes to what extent the system identified all the correct instances (i.e., recall is a measure of completeness). *F-measure* is an indicator which balances between *precision* and *recall* and gives a general estimation of how well the system performed. According to [21], recall is a measure of how complete the set of correctly identified instances is in relation to all the correct instances available; precision measures how correct is the set of identified instances. In simpler terms, a high precision means that the identified results were mainly correct, while a high recall means that most of the correct results were identified.

We might see the measures of precision, recall and F-measure as:

$$Precision = \frac{C}{A}, \tag{2.1}$$

being C the number of instances correctly identified by the system and A the number of instances available to choose from.

$$Recall = \frac{C}{A_C}, \tag{2.2}$$

being C the number of instances correctly identified by the system and A_C the number of all the correct instances.

$$F = 2 \times \frac{(Precision \times Recall)}{(Precision + Recall)}, \tag{2.3}$$

being F the F-measure we want to compute, and $Precision$ and $Recall$ the values computed in 4.1 and 4.2, respectively.

[44] propose the use of Active Learning (AL) in order to solve the problem of acquisition of annotated data for IE systems based on Machine Learning. The authors argue that AL have already been applied successfully to some NLP tasks such as part-of-speech (POS) tagging and text categorisation, but these are classification tasks in the sense one wishes to assign a class to the unclassified instances (e.g., a POS tag in the task of part-of-speech tagging and a category, such as "journalistic" or "medical", in the case of a text categorisation task).

The task of Information Extraction is not one of classification, like the two we have just mentioned, but could be seen as a complex task. The annotated data most likely to be provided by a human to an IE system is not simply a label or a category, but should rather be a filled template (i.e., a template filled

with the information one wants to extract). The RAPIER system, an IE system proposed by [44], has the example annotations provided by the user in the form of filled templates, not class labels. When the authors applied AL to RAPIER, the results consistently outperformed RAPIER alone – in order to achieve the same F-measure of 74.56, the usage of RAPIER with AL needed 150 examples, while RAPIER alone needed 270 examples.

[42] proposed WHISK, an IE system that can extract information from structured, semi-structured or free text. Structured text is, for instance, text within HTML tags that do not need linguistic parsing for its extraction. Semi-structured text are texts that follow some rules, but are commonly ungrammatical, such as rental ads or telegraphs. Free texts are texts written in natural language with no further restrictions, such as news articles.

WHISK's rules are like regular expressions and are learnt by applying supervised learning. The rules are induced from hand-tagged instances given by human annotators. In case of structured or semi-structured text, WHISK does not require any syntactic analysis. In case of free text, good results are obtained when its input undergo syntactic analysys and semantic tagging.

The author compares WHISK with several other older IE systems, and such as the Wrapper Induction system ([27] apud [42]), SRV ([16] apud [42]), RAPIER ([5] apud [42]), CRYSTAL ([41] apud [42]), FOIL ([35] apud [42]) or AutoSlog ([37] apud [42]), amongst others ([42]).

According to [42], IE systems might be a first step towards the discovery of trends in massive amounts of text data.

2.2.5 Information Extraction and Text Mining

According to [11], the field of Text Mining (TM) aims at extracting useful information from sets of unstructured documents. By unstructured documents we mean documents written in natural language. These systems are usually constituted of preprocessing routines, algorithms for dealing with pattern discovery, and a presentation layer containing visualisations for the data. The preprocessing routines are a set of tasks used to transform the unstructured text into an intermediate, explicitly structured representation. Some of these tasks might include part-of-speech tagging or syntactic parsing.

A part-of-speech (POS) tagger is a program that receives unstructured text as input and tags (assigns) each word of that text with part-of-speech tags. There are different sets of tags used by different POS taggers, but as an illustration, good examples of parts-of-speech tags are *noun*, *adjective* and *verb*. A syntactic parser is a program that, given some text written in natural language, outputs the syntactic representation (usually a syntactic tree) of that text. These two tasks are important for they provide metadata important to the process of pattern discovery that comes afterwards.

According to [11], the main task of text mining systems involve pattern discovery, trend analysis and incremental knowledge discovery, and the three most common patterns found in text mining are distributions (and proportions), frequent (and near-frequent) sets and associations. [8] states that the co-ocurrence-based methods look for concepts that occur in the same piece of text – be it a sentence, a paragraph or even a full abstract – and attempt at establishing a relationship between them. These

methods are fairly simple and were applied mostly in early works [3]. More advanced methods crafted for text mining are rule-based and machine learning methods.

In text mining, the "useful information" one wants to extract, mentioned earlier, is usually unknown. In biomedical text mining, examples of the extracted information might be new, undiscovered relationships between genes and diseases. In text mining applied to security applications, examples of information to extract could be posts related to a specific sensitive subject. The main difference between Information Extraction and Text Mining systems is the type of information one aims at obtaining.

In conclusion, text mining is more related to the *discovery* of *new* information from a set of unstructured documents. Information Extraction systems aim at the extraction of information already known in advance – for instance, a person's name, this person's given addresses or the companies he or she has worked in.

2.3 Related work

There are several different works that use Information Extraction or Text Mining techniques and use scientific papers as the main corpora they work upon. There are some proposals in the literature which are more or less related to our task, that is, the task of extracting information from online databases of scientific papers in Natural Language Processing publications.

Some authors use Information Extraction, such as Conditional Random Fields, and/or Machine Learning techniques, such as Hidden Markov Models, to extract portions of papers important to tasks such as field-based search, author analysis and citation analysis [32, 39]. There are also applications of Support Vector Machines for automatically extracting metadata from research papers [17].

There are other works available in the literature which also propose some kind of application of Information Extraction techniques on a corpus of scientific papers, but with a final objective clearly distinct from ours.

For instance, there is work on the automatic extraction of information from research papers with the aim of providing readers a comprehensive summary of the articles' contents [25].

Morevoer, there are several research groups in the field of bioinformatics and its interfaces with biomedicine and medicine applying Information Extraction and Text Mining on scientific research papers in biological and medical publications. Several authors report works on IE systems for aiding in the manipulation of medical and biomedical data [9, 21, 31]. These works won't be used in the evaluation of our system because their objectives are clearly distinct from ours, even though they use similar techniques in order to accomplish their results. For a survey on the existing methods, please refer to [20].

2.3.1 Seymore et al., 1999

[39] propose a method that uses Hidden Markov Models (HMMs) in order to extract information from headers of computer science research papers. The paper header is defined as everything that precedes the main body of the paper, or the end of the first page, whichever occurs first.

[3] In that case, early works in the field of biomedical text mining.

The authors extract the following information from the text ([39]):

- *title*: the title of the paper;
- *author*: the name of the authors;
- *affiliation*: the university or research institute the authors are affiliated to at the time of the publication release;
- *address*: the authors' given addresses;
- *note*: any footnotes found;
- *email*: email addresses of the authors;
- *date*: any dates found;
- *abstract*: the paper's abstract;
- *introduction (intro)*: the paper's introduction;
- *phone*: the authors' phone numbers;
- *keywords*: the topics analysed in the document;
- *website*: URL of authors' webpages;
- *degree*: language associated with thesis degree;
- *publication number*: document's publication number;
- *page*: the end of the page.

Even though the authors extract the paper's abstract, introduction and page, these are not part of their method evaluation.

[39] build several models with different parameter settings. The models are either trained on unlabeled, labeled or distantly-labeled data. The distantly-labeled data are, according to the authors, data which has been labeled for a different purpose but can nevertheless still be applied – it might happen that only a part of the labels are relevant. Examples of these are Bibtex files, which contain fields such as title and authors but also contain several other fields that does not concern the task.

The results computed by [39] consists of word classification accuracy, measured by the percentage of the words correctly labeled by the HMM. They report results for each class separately and also for the whole task, for which the overall accuracy reported is 90.1%.

2.3.1.1 Han et al., 2003

[17] propose a method for the automatic extraction of metadata from research papers. They use Support Vector Machines (SVMs) and, like [39], also extract information from the paper header – which is again

defined as the portion of the paper that precedes its main body. The information extracted by [17] is the same information as the one presented by [39].

[17] extract word and line-specific features for their SVM classifier. They use a rule-based heuristic in order to cluster words together, for the case of multi-word units, and also external databases (resources) for training their word-classifiers – such as a collection of first and last names, month names and abbreviations, country names, amongst others. For their line-specific features, they include information such as the quantity of words a line contains, the line number, amongst others.

They evaluate their work using precision, recall, F-measure and accuracy. Additionally, [17] compare their results against the results reported by [39] and state that they achieve better results then HMM-based methods (for which [39] have the best results up until the publication of their paper).

The results reported are an overall accurary of 92.9%. The precision, recall and F-measure are computed only for each class (title, author, affiliation, etc.). They do not explicitly report, therefore, their method's overall precision, recall and F-measure. The F-measures reported for some of the fields are:

- *title*: F-measure $= 89.3\%$;
- *author*: F-measure $= 91.8\%$;
- *affiliation*: F-measure $= 88.7\%$;
- *address*: F-measure $= 89.3\%$.

2.3.1.2 Besagni et al., 2003

[4] present an Information Extraction method for the automatic extraction of information from the references of scientific papers from journals in pharmacology. They extract the following information: the paper's title, the journal name, the publication date, the journal volume, the pagination of the paper within the journal and the authors' names.

Amongst other numbers, they report the following results:

- *authors*: accuracy $= 90.2\%$;
- *paper title*: accuracy $= 82.4\%$;
- *journal name*: accuracy $= 92.4\%$;
- *date*: accuracy $= 97.7\%$.

2.3.1.3 Peng and McCallum, 2004

[32] propose an Information Extraction system to extract what they call research papers' metadata – such as title, author, and institution. They extract that information from both the paper header and its references – as opposed to [17] and [39], which only use the paper header. The header is defined the same way as [17] and [39], namely as the portion of the paper that comes before the beginning of its first section or its main body. The information extracted is also the same data extracted by [17] and [39].

The method is based on the theory of Conditional Random Fields [28] (CRF), which are a way for building probabilistic models tailored for segmenting and labeling of sequencial data. According to [28], CRFs aims at combining the benefits of conditional models, such as Hidden Markov Models, with the benefits of Random Field models.

The overall accuracy of the method proposed by [32] for extractions from the paper header is 98.3% and the average F-measure is 93.9%. The authors extract the following fields: title, author, affiliation, address, note, email, date, abstract, phone, keywords, website, degree, publication number. The authors also report numbers for extractions from the paper references, which are accuracy of 95.37% and F-measure of 91.5%.

The results obtained by these authors are, according to themselves, significantly better than the state-of-the-art Hidden Markov Models (HMMs) and Support Vector Machines (SVMs), setting a new state-of-the-art performance for the task. [32] remarks that their so-called "layout features" (whether a word is in a beginning of a line, in the middle of a line of in the end of a line) are responsible for raising the F-measure from 88.8% to 93.9%.

2.3.1.4 Powley and Robert, 2007

[33] present a method for high accuracy citation and author name identification. They use both in-text citations and the references section of the papers in order to extract the bibliographic information they are interested at.

They use as their corpus reserach papers downloaded from the ACL anthology website, as we do in the IEforNLP system. Since the ACL anthology papers are only available in PDF, they use PDFBox [4] to convert the PDFs into a text-only format, as we also do in the IEforNLP system.

The authors conduct experiments on their method on a subcorpus specially chosen for that task – which was not previously "seen" in the method's development process.

As we have already mentioned, [33] extract citation information and use in-text citations as well as the references section of the papers. They extract the in-text citations and the information from the references section in one integrated process. Amongst other numbers, they report an accuracy of 94.85% for the citation-reference matching – that is, the amount of times their method correctly matched a given in-text citation to its entry in the references section.

[4] They provide the website address of http://www.pdfbox.org/ for information on PDFBox, which does not point to the PDFBox website anymore. The software is now under the Apache Software Foundation and is available at http://pdfbox.apache.org/ Last access on March 14, 2013.

Chapter 3

The IEforNLP system

In this chapter, we propose the system **IEforNLP** (**I**nformation **E**xtraction **for** **N**atural **L**anguage **P**rocessing). This system can be seen as an Information Extraction system and aims at the extraction of specific information from online databases of scientific papers in Natural Language Processing.

In section 3.1, we present the corpus adopted for this work and discuss the reasons for the choices we made. In section 3.2, we present the architecture of the system IEforNLP in a systemic way and we also explain its functionalities by describing how each of its components work and interact with each other. Moreover, we also discuss some implementation details and the solutions proposed for the problems we encountered.

3.1 Corpus

The first important choice to make is the decision of which publications – journals, conferences and workshops – shall be part of our experiment and which shall not. We believe, first, that the list of publications to be analysed must be scientifically sound. We also believe the list should encompass research being conducted in all the best research centres in the world (i.e., it should be comprehensive, geographically representative). Finally, we aim at creating a list which researchers should (at least theoretically) agree upon as both scientifically sound and comprehensive. According to [2], "The Association for Computational Linguistics is THE international scientific and professional society for people working on problems involving natural language and computation." The ACL provide free access to the contents of some of the most important publications in NLP world-wide [1].

In order to better choose the conferences, journals and workshops that integrate our list of publications, we developed some simple "rules" to drive us towards scientific soundness and comprehensibility.

The first rule for choosing a conference, journal or workshop to be a part of the list of publications used by IEforNLP is, therefore, whether this publication is available under the ACL umbrella – the reason being what we have already pointed out about the ACL. As the ACL is undisputably the most

[1] See the anthology in http://www.aclweb.org/anthology-new/ Last access on March 14, 2013

important forum for researchers in Natural Language Processing, we believe restricting our search to the publications under its umbrella is a form of assuring the quality of the publication.

According to [13] in a paper recently published in the Annual Meeting of the Association for Computational Linguistics in 2011, the conferences ACL, COLING, LREC, EMNLP, MT Summit and the journal CL are some of the the main journals and conferences on speech and language in the field of NLP. Given the information already discussed above, the full list of publications included for IEforNLP is given below.

- The Computational Linguistics Journal [2] (CL)
- The Annual Meeting of the Association for Computational Linguistics [3] (ACL)
- Conference on Empirical Methods in Natural Language Processing [4] (EMNLP)
- International Conference on Computational Linguistics [5] (COLING)
- International Conference on Language Resources and Evaluation [6] (LREC)
- Machine Translation Summit [7] (MTSUMMIT)
- European Chapter of the ACL Conference [8] (EACL)
- Conference of the North American Chapter of the Association for Computational Linguistics [9] (NAACL)
- International Workshop on Semantic Evaluation [10] (SEMEVAL)
- Applied Natural Language Processing Conference [11] (ANLP)
- Conference of the North American Chapter of the Association for Computational Linguistics: Human Language Technologies [12] (HLT)
- Pacific Asia Conference on Language, Information, and Computation [13] (PACLIC)

We added to our publication's list, in addition to the publications already chosen, the conferences and workshops EACL, NAACL, SEMEVAL, ANLP, HLT and PACLIC. In order to assure that all regions of the globe be represented, we chose conferences under the ACL from the following regions: Europe (EACL), North-America (NAACL and HLT), and the Pacific-Asia region (PACLIC). The other missing

[2] See http://www.mitpressjournals.org/loi/coli Last access on March 14, 2013.
[3] See http://acl2013.org/site/ Last access on March 14, 2013.
[4] See http://emnlp-conll2012.unige.ch/ Last access on March 14, 2013.
[5] See http://www.aclweb.org/index.php?option=com_content&task=view&id=93&Itemid=26 Last access on March 14, 2013.
[6] See http://www.lrec-conf.org/ Last access on March 14, 2013.
[7] See http://www.mtsummit2013.info/ Last access on March 14, 2013.
[8] See http://www.eacl.org/ Last access on March 14, 2013.
[9] See http://naacl2013.naacl.org/ Last access on March 14, 2013.
[10] See http://www.cs.york.ac.uk/semeval-2013/ Last access on March 14, 2013.
[11] See http://www.informatik.uni-trier.de/~ley/db/conf/anlp/index.html Last access on March 14, 2013.
[12] See http://aclweb.org/anthology-new/N/N12/ Last access on March 14, 2013.
[13] See http://paclic26.cs.ui.ac.id/ Last access on March 14, 2013.

conferences, SEMEVAL and ANLP, were chosen for being international and focused on evaluation, and for representing applied research, respectively.

After we decided which journals, conferences and workshops shall be part of the publication list to be used within IEforNLP, we must define a time frame to work with. Some of these publications have been around for a long time already. The *Computational Linguistics* journal have links to the proceedings of the years as far as the early 1970s, the Annual Meeting of the ACL also. Some have proceedings only for the late 1990s, or from the year 2000 on, such as SEMEVAL, EMNLP and NAACL. Taking that into consideration, we thus defined our time frame as emcompassing the last 10 years, starting from 2003 until 2012. We believe a 10-year time frame allows us to work with a reasonably large period without having to deal with low-quality papers converted from OCR (as have been noted by [33]).

3.2 Architecture

The system IEforNLP is composed of four main components, as illustrated in Figure 3.1: a crawler, a pre-processor, an Information Extraction component and a database persistance component. Execution units or components are shown in Figure 3.1 as boxes with continuos lines. The execution flow is shown in Figure 3.1 in black continuous lines. When there is a black arrow from one component to another, this means that after the execution of the first component, the next component starts executing. The information flow between the components can be seen in Figure 3.1 in blue dotted lines. A blue dotted line connecting one component to another means, thus, that the first component provides (outputs) information that becomes input to the next component. We see in Figure 3.1 that the Information Extraction component is the only one which receives information from two other components – the crawler and the pre-processing.

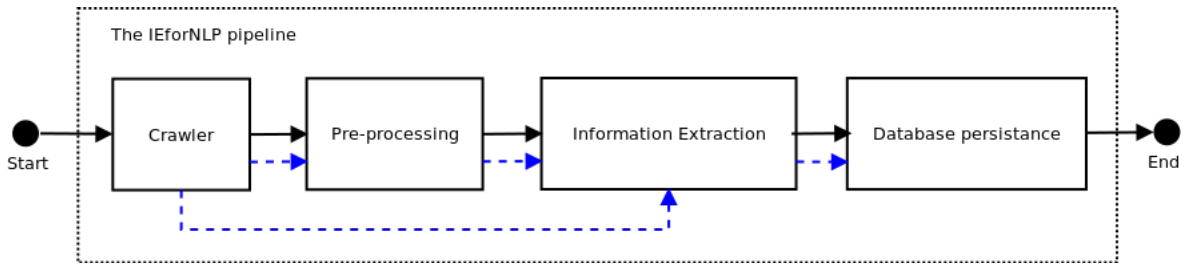

Figure 3.1: The pipeline of the IEforNLP system.

The crawler is responsible for the download of papers from selected conferences, workshops and journals. It is also responsible for extracting information that will be used later on to aid in the extractions made by the Information Extraction component. The pre-processor is responsible for converting the downloaded papers [14] into a text-only format suitable for the Information Extraction component. The IE component, as its name already implies, is responsible for the extraction of information from the pre-processed research papers. This information is then handed in to the database persistance component, which saves everything into a relational database.

[14]The papers downloaded are available under the PDF format.

We shall now define the four components of our architecture pipeline: the crawler, the pre-processor, the IE component and the database persistance component.

3.2.1 Crawler

A crawler, also called a web crawler or a web spider, is a piece of software that automatically downloads websites for a given purpose. This purpose might be, for instance, the creation of indexes for these websites to be used by search engines, or even to automatically build backups for them. A crawler usually starts its downloading task with an initial list of URLs called its *seed*. It might then add entries to this initial list by the automatic extraction of hyperlinks from its seed's webpages.

The crawler we envisage downloads all the papers published in a list of journals, conferences and workshops, for a given time frame. These papers are downloaded from the Association for Computational Linguistics (ACL) anthology website [15], which is the only URL in our crawler's *seed*.

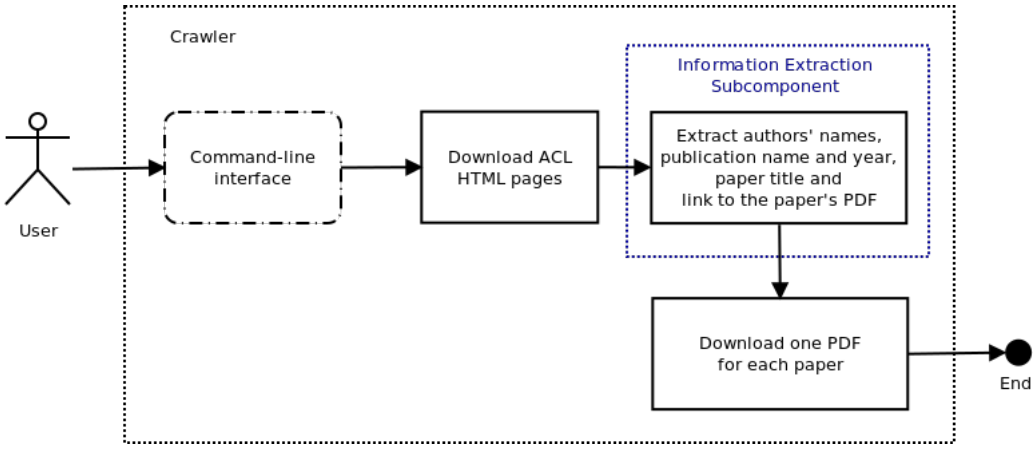

Figure 3.2: The architecture of the crawler used by the IEforNLP system.

The schematic architecture of our crawler is illustrated in Figure 3.2. The crawler must be provided with an input list of the publications to download – the name of the journals, conferences and workshops included – and also the time frame to consider – which publications' years should be included and which should not. We use a command-line interface to input this data to the crawler, as can be seen in the box the user interacts with.

With this information, the crawler then parses the HTML file of the ACL website (our seed) and extracts some links from it. These are the links for the HTML websites that contain the actual links to the papers' PDFs. We shall refer to these links or websites as *second degree* websites. In the ACL website, there is one HTML website for each publication year. This means that there is, for instance, one link for the page containing the proceedings of the Computational Linguistics (CL) journal for the year of 2012, another year for the CL journal for the year 2011, and so on. The same happens for all other ACL publications considered.

After the first execution unit [16], *Download ACL HTML pages*, described in Figure 3.2 is finished a list

[15] http://www.aclweb.org/anthology-new/ Last access on March 14, 2013.

[16] One execution unit is represented by a rectangular box with a continuous line.

with the downloaded second degree websites, which contains one entry per publication per year, is created.

Each entry of this list, which is an HTML website, will contain the proceedings of a publication for a given year. These proceedings always contain links to one PDF for each paper included in it, and might also contain one link to a PDF containing all the papers altogether (the full proceedings of that publication for that year). In addition to links to the PDF files, these second degree HTML websites also contain the names of each of the paper's authors, the paper title and the publication name and year.

The second execution unit, *Extract authors' names, publication name and year, paper title and link to paper's PDF*, described in Figure 3.2, is considered to be an Information Extraction subcomponent, as can be seen by the blue dotted box. After this execution unit is finished, we will have a list containing, for each paper in the proceedings: *(i)* one link for the PDF file containing the contents of the paper; *(ii)* the authors of the paper; *(iii)* the title of the paper; *(iv)* the title of the publication; *(v)* the year of the publication. While the crawler extracts some information as well as the Information Extraction component, it only does so directly from the ACL website's HTML. The handling of the actual research papers is done only by the Information Extraction component.

Finally, given the list of links to PDF files available, after the third execution unit, *Download one PDF for each paper*, described in Figure 3.2 is finished, we will have a list of PDFs downloaded, one for each paper in the proceedings. It is important to state that we also keep the other information numbered *(i)-(v)* for each of the PDFs downloaded as this information will be used afterwards in the pipeline by the Information Extraction component.

3.2.2 Pre-processor

A pre-processor is responsible for tranforming input data in different ways so that this data fits the requirements of the program that will use it afterwards. Our pre-processor is responsible for converting PDFs into a text-only format, splitting the text into arbitrary sections and fixing its contents when necessary.

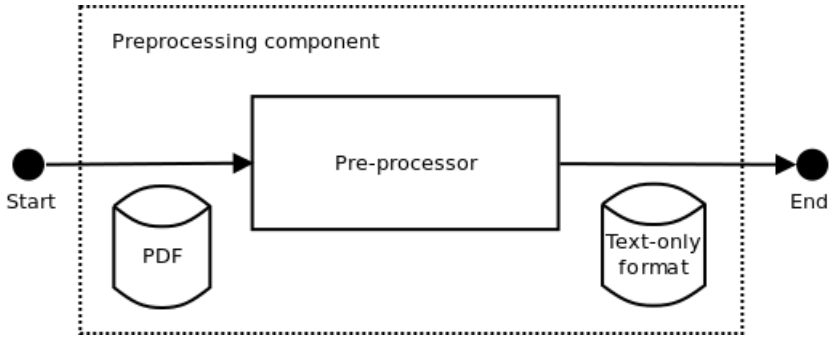

Figure 3.3: The pre-processor component of the IEforNLP system.

A general-purpose pre-processing unit for Natural Language Processing might perform several other tasks apart from PDF convertion, such as tokenisation, part-of-speech tagging, synctactic parsing, shallow parsing or deep parsing. Nevertheless, for the purposes of this work we chose to use a text only

representation without these added information.

The only information added to the text is the splitting. Our pre-processor splits the text into the following sections: header, abstract, keywords and body. These sections will be used afterwards by the Information Extraction component, to guide its extraction processes.

In Figure 3.3, we find the schematic structure of the pre-processor component of the IEforNLP system. In order to perform the PDF convertion task, we utilize a third-party software, Apache PDFBox [17]. The Apache PDFBox is an open source tool written in Java and distributed under the Apache Software Foundation, an open source community acknowledged world-wide.

The first and only execution unit [18] described in Figure 3.3 is executed upon each and every PDF file we have downloaded with our crawler, in the previous stage. After the first execution unit is finished a text-only representation of the PDF file is created, and that textual representation is splitted into header, abstract, keywords and body.

Sometimes it is necessary to fix the contents of the converted, text-only files generated. These fixes are usually simple, such as putting words splitted at the end of lines together or removing any mathematical symbols that might have been part of the text. This is done in order to prevent interference with the next stages in the pipeline. In order to do so, we use regular expressions.

3.2.3 Information Extraction

The Information Extraction component (IEC) is responsible for extracting information from scientific papers texts converted from PDF.

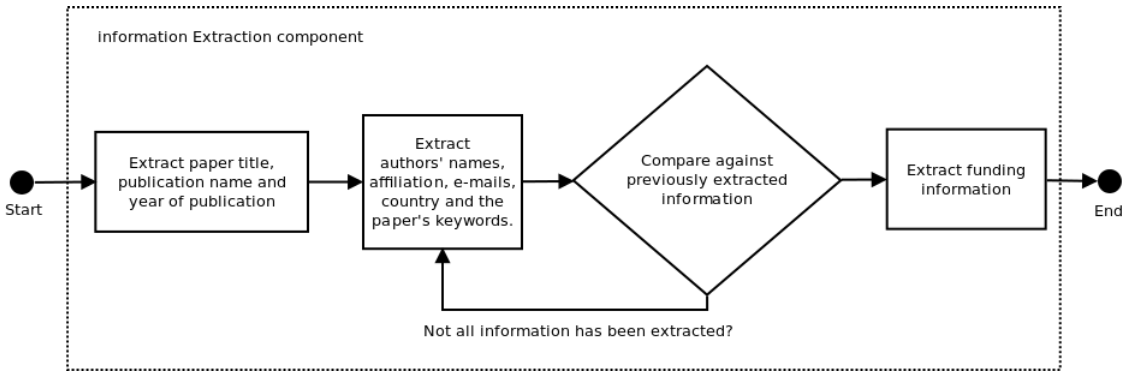

Figure 3.4: The architecture of the Information Extraction component used by the IEforNLP system.

The schematic architecture of our IEC is illustrated in Figure 3.4. The IEC receives as input the output of the pre-processor component (i.e., a text-only file converted from the PDF containing the contents of a research paper) and also information extracted by the crawler (i.e., the publication name and year, the paper title and the paper's authors).

We define an unary field as a field that corresponds to exactly one entry or term extracted (this *one entry* is always measured in relation to the research paper). Additionally, we define an n-ary field as a field that might have one or more corresponding entries or terms extracted from one paper. Examples of

[17] http://pdfbox.apache.org/ The software is available under the Apache License v2.0. Last access on March 26.

[18] One execution unit is represented by a rectangular box with a continuous line.

unary fields are the paper title and the publication name. Examples of n-ary fields are authors' names. After the first execution unit [19] described in Figure 3.4 is finished an entry with the unary fields extracted from the research paper is created. The unary fields extracted are the paper title, publication name and publication year.

After the second execution unit described in Figure 3.4 is finished, we shall have the n-ary fields extracted from the research paper text. The n-ary fields extracted are the paper's keywords, the paper's authors, the university or research centre of affiliation for each of the authors, the authors' emails and the country the university or research centre is a part of.

Right after the extraction of the n-ary fields, we decide whether the fields extracted by the IEC up to that point match the information extracted by the crawler [20]. If the information does not match, we use the information obtained from the crawler as input to the second execution unit and re-execute it.

The second execution unit, given the information extracted by the crawler, will attempt at matching the information by using certain rules and regular expressions. If even after this second attempt the second execution unit cannot find a match between the information extracted from the crawler and from the IEC, it keeps the information extracted by the crawler, saves the information extracted by the IEC for auditing purposes and flags the entry with the flag *unmatched*. On the other hand, if the information extracted by the crawler and the IEC match, we keep the most descriptive entry [21], do not flag the entry with anything and continue to the next execution unit.

Finally, after the second execution unit has finished being executed, we then follow the pipeline and execute the third execution unit described in Figure 3.4. After this third execution unit is finished, we will have extracted information on the funding. This information is either extracted from the *Acknowledgements* section or from a footnote in the first page of the paper, and by means of pattern matching with regular expressions.

After the funding information has been extracted, the execution of the IEC is finished and we then follow to the next component in the pipeline. If there is no funding information available in the research paper – or no such information could be found –, the paper entry is flagged as *no funding*. The pipeline execution then follows to its next and final stage.

3.2.4 Database Persistance

The Database Persistance component is responsible for saving the information automatically extracted into a relational database for further use. The relational database management system (DBMS) we use is MySQL [22] version 5.5.29. In Figure 3.5, we show the entity-relationship model (ER model) used for storing all the information extracted from the research papers.

[19] One execution unit is represented by a rectangular box with a continuous line.

[20] At that point, a match is an exact, non-case-sensitive string match. For instance, *John Smith* matches *john smith*, but *University of London* does not match *London City University*.

[21] Let's say, for example, the crawler extracted the names *John Smith* and *M.Russell*, and the IEC extracted *J.Smith* and *Mark F. Russell*. The first matching attempt fails, but the second attempt will match the two entries (one of the several regular expressions for name matching is activated). The most descriptive entries, in that case, are *John Smith* and *Mark F. Russell*.

[22] http://dev.mysql.com/downloads/mysql/ Last access on March 14, 2013.

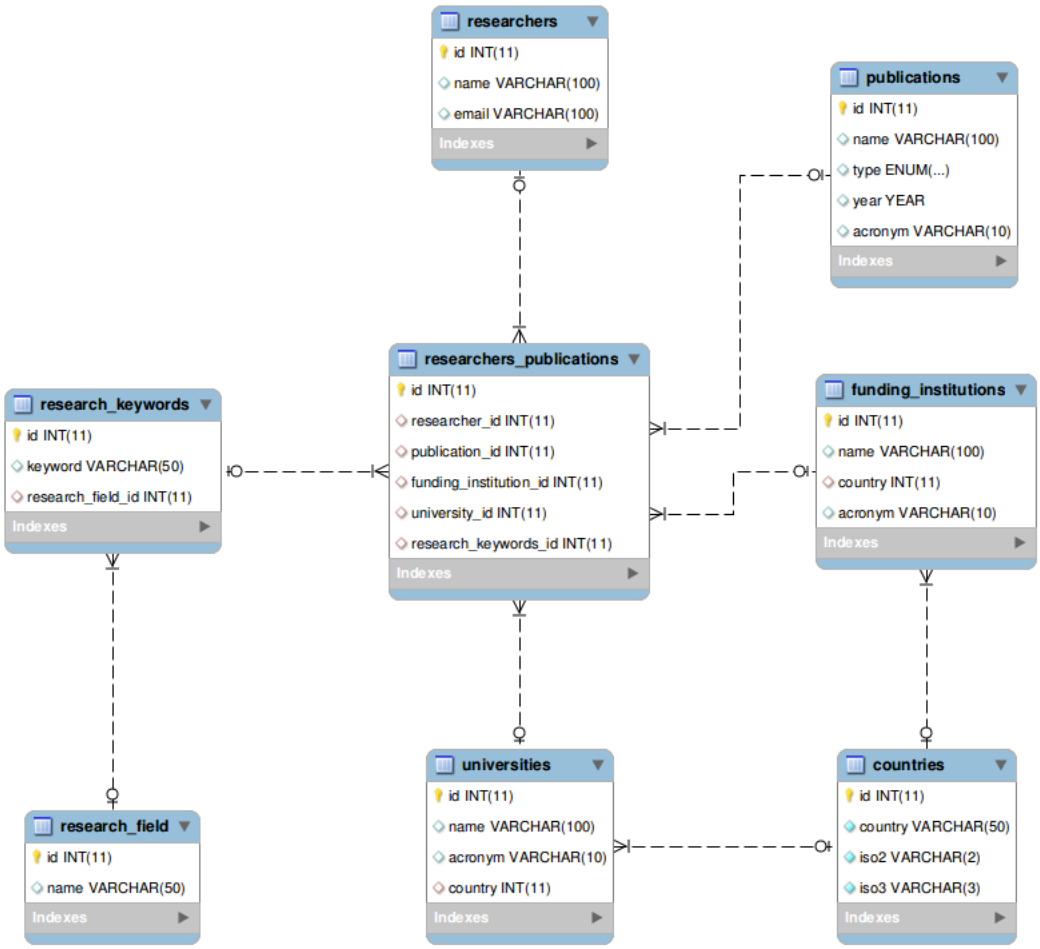

Figure 3.5: The Entity-Relationship model of our database.

We draw upon the theory of data warehousing and use a star schema with seven *dimension* tables and one *fact* table. For more details on database normalisation, data warehousing and the star schema, please refer to [3, 18, 26]. As we see in Figure 3.5, there are eight tables in total, seven dimensions and one fact:

1. dimension *researchers*, which contains a list of the authors, with their names and email addresses;
2. dimension *universities*, which stores the university name, the university acronym (if one exists) and a reference to the country the university belongs to;
3. (sub)dimension *countries*, which contains a list of the countries of the world, along with their ISO codes [23];
4. dimension *funding_institutions*, which keeps information of the different funding institutions. The information kept is the funding institution name, its acronym (if one exists) and a reference to the country it belongs to.
5. dimension *publications*, which stores the publication name, type (whether it is a journal, a conference or a workshop), year of publication and acronym (if one exists);
6. dimension *researcher_keywords*, which contains a list with the papers' keywords;
7. dimension *researcher_field*, which contains a list with the papers' main research fields within the NLP research environment;
8. fact *researchers_publications*, which holds the information of which project(s) the funding refers to, in which funding institution, and for which publication – the publication has also a reference to the researchers who authored it, and to which research institution they were affiliated to at the time of the publication).

These six tables store all the information we extract from the research papers. The structure of the tables is not normalised and it has been designed in that way in order to facilitate the access of future applications that might use this information. This table structure is used when one wishes to design a database suitable for queries about our facts. We shall retrieve our facts and parametrise its queries using the existing dimensions.

With a database structure such as this one, we can build efficient queries using the Structured Query Language (SQL) in order to build queries of the following type:

1. How many different universities, represented by their researchers, have published research papers in NLP?
2. What is the percentage of the NLP publications published by Europeans, North-American or Asian research institutes and/or universities?

[23]For more information on the country list and their ISO codes, please refer to http://www.iso.org/iso/country_codes Last access on March 14, 2013.

3. Researchers from which universities in the world have published at least a certain number of papers in one research field of interest?

3.2.5 Implementation Details

We shall now discuss some of the implementation details of the system IEforNLP. The system was implemented using different programming languages (more specifically, scripting programming languages), which were connected afterwards using Shell scripts [24]. These Shell scripts were responsible for the execution of the different components in the correct order and also for providing the information necessary for each of them to run. The scripts were written in Shell script (all the script files ending in *.sh*) and Python [25] version 2.7 (all the script files ending in *.py*), and we also used a third-party library written in Java (already mentioned in Section 3.2.2) in order to convert PDFs into machine-friendly text files.

We now present the different scripts, written in the programming languages Python and Shell script, and relate them to the execution units (components) described in Figure 3.1. There is a script named *execute_scripts.sh* which is responsible for the execution of all the other scripts, in the right order. It contains a list with the publication names (journals, conferences and workshops) and also a list with the publication years we are interested at. It is responsible for providing all the necessary inputs for each of the invoked scripts, managing the information flow between them.

Here follows the scripts invoked by *execute_scripts.sh*, in order of invokation (and consequently execution):

(i) *nlp-html-crawler.py*;

(ii) *nlp-pdf-crawler.py*;

(iii) *nlp-download-pdf.py*;

(iv) *nlp-convert-pdfs-to-text.py*;

(v) *nlp-extract-information-from-texts.py*;

(vi) *nlp-persist-database.py*.

Before we explain what each script does, it is important to state that there is one script that can be invoked by the scripts enumerated. It is the *download-file.sh*, which is a script that simply downloads files and places them in the right folder.

The first three scripts invoked (*nlp-html-crawler.py*, *nlp-pdf-crawler.py*, and *nlp-download-pdf.py*) are represented in Figure 3.1 as the first execution unit (i.e., the Crawler). The *nlp-html-crawler.py* parses the ACL anthology website and invokes *download-file.sh* to download each of the HTML files containing links to the corresponding publications' PDFs we are interested at. The *nlp-pdf-crawler.py* parses these HTML files (downloaded by proxy by *nlp-html-crawler.py*) and extracts the links to the actual PDF files. The *nlp-download-pdf.py* then also invokes *download-file.sh* and downloads all the PDFs (identified by *nlp-pdf-crawler.py*), placing them in appropriate folders.

[24] http://tldp.org/LDP/abs/html/index.html Last access on March 14, 2013.
[25] http://www.python.org/ Last access on March 14, 2013.

The *nlp-convert-pdfs-to-text.py* is the pre-processor component, represented in Figure 3.1 as the second execution unit. As described in Section 3.2.2, it uses the third-party software PDFBox in order to convert the PDFs just downloaded into a text-only, machine-friendly format. Additionally, it splits the text into four sections (header, abstract, keywords and body) and also applies some minor corrections to it.

The *nlp-extract-information-from-texts.py* is the Information Extraction component, represented in Figure 3.1 as the third execution unit. Its functionalities have been extensively examined in Section 3.2.3.

Finally, the script *nlp-persist-database.py* stores all the information extracted in the previous steps into the MySQL DBMS. The database ER model is thouroughly described in Section 3.2.4.

This ends the architecture of the IEforNLP system proposed in this work. We shall now evaluate our system and provide the obtained results for analysis.

Chapter 4

Evaluation

In this work we propose an Information Extraction system that extracts information from Natural Language Processing online databases of scientific publications, and also a database which is generated based on this information. The objective of our evaluation is to analyse how well the system performs in three different stages:

(i) the quality of the Crawler, that is how well does the crawler download only and all the correct papers for the chosen NLP scientific publications.

(ii) the quality of the Information Extraction Component (IEC) of the system IEforNLP when applied to NLP scientific research papers.

(iii) the quality of the final information saved into our relational database. This information is the combination of the information obtained by the Crawler and the IEC and is **not** restricted to the texts of the papers. In other words, it includes all the fields extracted by the Crawler from the ACL webpages as long as the fields extracted by the IEC.

In order to answer *(i)*, we manually built a list with all the papers that should be downloaded, for each publication. We then compare the list of publications that our crawler have automatically downloaded against this list in order to measure its adequacy.

In order to answer *(ii)*, we perform two distinct evaluations, one using an *in-domain corpus* and another an *out-of-domain corpus*. We define our *in-domain corpus* as the one containing all the scientific research papers for the publications we have defined in 3.2 and within the time frame defined. We later define a subset of that corpus as our test set. We compute precision, recall, F-measure and accuracy for the task of extracting the fields defined in 1.2.1 when applied to this test set.

A second evaluation concerns how well does the system generalise if applied to a corpus other than the *in-domain* corpus we have defined. We define thus an *out-of-domain* corpus so that we may hypothesise about the application of this system on unplanned, different corpora. Nevertheless, it does not make sense to apply IEforNLP on any research papers from any field, without restricting the possibilities in some way. In order to do that, we then define our *out-of-domain* corpus as consisting of:

- research papers published in a conference, workshop or journal that is within the list of publications used in IEforNLP (see 3.2), but in a year not within the time frame defined for our *in-domain* corpus;

- research papers published in any conference, workshop or journal under the Association for Computational Linguistics (ACL) umbrella and available in the ACL website, with no restrictions on the time frame.

In other words, another way of defining our *out-of-domain* corpus is as consisting of all research papers in all of the journals, conferences and workshops under the ACL umbrella and available in the ACL website, except for the ones that have already been included in our *in-domain* corpus.

Finally, in order to answer *(iii)*, we also perform two distinct evaluations, one with an *in-domain* and another one with an *out-of-domain* corpus. These corpora are exactly the same ones defined for evaluating case *(ii)*.

4.1 Methodology and Results

As previously mentioned, we shall evaluate the crawler component, the IE component and the final information stored in our relational database separately. In order to evaluate the crawler, we assess how well does the crawler download only and all the correct papers for the chosen NLP scientific publications. In order to evaluate the quality of the information extraction component of the IEforNLP system, we apply it to two corpora: an *in-domain* corpus and an *out-of-domain* corpus. In order to evaluate the quality of the information saved into our relational database, we use the same *in-domain* and an *out-of-domain* corpora used in the previous evaluation.

4.1.1 Crawler

We decided to evaluate the Crawler on a small test set of research papers not involved in the IEforNLP development process. In order to do so, we chose the proceedings of 2012 of the journal *Computational Linguistics* and also the proceedings of 2012 of the Annual Conference of the ACL. There are 227 papers chosen for the Annual Conference of the ACL and 40 papers chosen for the journal *Computational Linguistics*, totalling 267 papers in the test set.

In order to evaluate the Crawler, we use the measures already discussed above. The difference is the way we define A, B, C and D. We define A as the number of research papers which were downloaded by the Crawler and were part of the list of research papers to be downloaded; B as the number of research papers which were downloaded by the Crawler and were not part of the list of research papers to be downloaded; C as the number of research papers which were not downloaded by the Crawler and were part of the list of research papers to be downloaded; and D as the number of research papers which were not downloaded by the Crawler and were not part of the list of research papers to be downloaded. The definitions of *precision*, *recall*, *F-measure* and *accuracy* are the same as the ones given in

Equations 4.1, 4.2, 4.3 and 4.4.

Precision	Recall	F-measure	Accuracy
1	1	1	1

Table 4.1: Statistical measures computed for the Crawler

In Table 4.1, we have the results of the statistical measures computed for the Crawler evaluation. As we can see, the Crawler correctly identifies and downloads all (and only) the 267 papers.

4.1.2 Information Extraction Component

The statistical measures computed are *precision*, *recall*, *F-measure* and *accuracy*. We compute these statistical measures because they are widely adopted by researchers for evaluating Information Extraction systems (as can be seen in [17, 25, 32, 39]), and so they make it possible to compare our work to the other works in the state-of-the-art review. Following these authors, we define A the number of true positive instances which were predicted as positive; B the number of true positive instances which were predicted as negative; C the number of true negative instances which were predicted as positive; and D the number of true negative instances which were predicted as negative.

$$Precision = \frac{A}{A + C} \tag{4.1}$$

$$Recall = \frac{A}{A + B} \tag{4.2}$$

$$F\text{-}measure = \frac{2 \times Precision \times Recall}{Precision + Recall} \tag{4.3}$$

$$Accuracy = \frac{A + D}{A + B + C + D} \tag{4.4}$$

We decided to evaluate the IEC on a small test set of research papers not involved in the IEforNLP development process. Following [33], this subcorpus comprised of 60 papers which were randomly selected from the publications already part of the *in-domain corpus*. From now on, we shall refer to this test set as *TS1*.

In Table 4.2, we have the statistical measures on *TS1* computed separately for each field extracted by the Information Extraction Component of the IEforNLP system.

Field	Precision	Recall	F-measure	Accuracy
Authors names	72.1%	78.3%	75.1%	60.1%
Affiliation	88.6%	88.6%	88.6%	79.6%
Country	75.9%	89.1%	81.9%	69.5%
Keywords	90.9%	90.0%	90.4%	82.6%
Funding information	85.7%	80.0%	85.8%	70.6%

Table 4.2: Statistical measures computed for *TS1*, for each field of the Information Extraction Component

In Table 4.3, we see the overall results computed for the Information Extraction Component, also using *TS1* as our corpus.

Precision	Recall	F-measure	Accuracy
82.6%	85.2%	84.4%	72.5%

Table 4.3: Overall statistical measures computed for *TS1*, for the Information Extraction Component

We conducted one more evaluation of the IEC, this time to measure how well it generalises on different corpora. As already mentioned above, this evaluation is conducted on a subset of the *out-of-domain corpus*. To measure how well does the system perform on this *out-of-domain* corpus, we compute the same *precision*, *recall*, *F-measure* and *accuracy* for the task of extracting the fields defined in 1.2.1, this time applied to a subset of our *out-of-domain* corpus. We choose, in total, three publications amongst the ones available under the ACL umbrella and which are not already part of the *in-domain* corpus, for the same time frame we have already adopted for our *in-domain* corpus. These publications are:

- Linguistic Annotation Workshop [1] (LAW)
- Annual Conference of the International Speech Communication Association [2] (INTERSPEECH)
- International Conference on Acoustics, Speech, and Signal Processing [3] (ICASSP)

From these three publications, we choose 60 papers randomly, the same way as we did for *TS1*. These 60 papers of these three publications comprise the subset of our *out-of-domain* corpus used as our second

[1] See http://www.aclweb.org/anthology-new/W/W12/#3600 Last access on March 14, 2013.
[2] See http://www.interspeech2013.org/ Last access on March 14, 2013.
[3] See http://www.icassp2013.com/ Last access on March 14, 2013.

test set. From now on, we shall refer to this test set as *TS2*. In Table 4.4, we have the statistical measures on *TS2* computed separately for each field extracted by the IEforNLP system.

Field	Precision	Recall	F-measure	Accuracy
Authors	67.6%	69.7%	68.6%	52.2%
Affiliation	73.8%	86.9%	79.8%	66.4%
Country	63.9%	81.2%	71.5%	60.5%
Keywords	77.8%	78.4%	78.1%	74.6%
Funding information	77.1%	72.5%	74.7%	59.7%

Table 4.4: Statistical measures computed for *TS2*, for each field of the Information Extraction Component

In Table 4.5, we see the overall results computed for the Information Extraction Component, this time using *TS2* as our corpus.

Precision	Recall	F-measure	Accuracy
72.04%	77.74%	74.54%	62.68%

Table 4.5: Overall statistical measures computed for *TS2*, for the Information Extraction Component

4.1.3 Database

In order to evaluate the information saved into our relational database, we use the same statistical measures computed for the previous evaluations: *precision*, *recall*, *F-measure* and *accuracy*. For the sake of brevity, we decided to use one of the test sets we used for the evaluation of the IE component, namely *TS1*.

TS1 consists of 60 randomly chosen papers, which were downloaded from one of the publications defined in Section 3.1. We computed statistical measures separately for each of the fields extracted, and also the overall results obtained, as can be seen in Table 4.6.

4.2 Final Comments

We can see from the results obtained on the evaluation that, first, the crawler works exactly as expected; second, that even though the IEC does not perform so well, it achieves reasonable overall results, as

Field	Precision	Recall	F-measure	Accuracy
Publication name	100%	100%	100%	100%
Publication year	100%	100%	100%	100%
Authors	99.2%	99.2%	99.3%	98.6%
Affiliation	92.5%	89.1%	90.8%	83.1%
Country	86.6%	94.2%	90.2%	82.2%
Keywords	90.9%	90.0%	90.4%	82.6%
Funding information	85.7%	80.0%	85.8%	70.6%
Overall results	93.6%	93.2%	93.8%	88.9%

Table 4.6: Statistical measures computed for *TS1*, for the final information stored in our relational database

indicated by the F-measure of 84.4%; finally, that the final quality of the information saved into the database is very good, as measured by the overall F-measure of 93.8%.

Additionally, the results of the evaluation of the IEC on *TS1* and *TS2* were as expected. *TS1* had better results then *TS2* in all separate fields, and consequently in the overall results as well. The smaller result computed for *TS2* was a F-measure of 68.6% for the field author name. The smaller result computed for *TS1* was also for the field author name, but this time a slightly higher F-measure value of 75.1%.

The information that is actually saved into the database has 100% F-measure for publication name and year, which were part of the information obtained by the crawler directly from the ACL website. Also, the authors' names are extracted by the crawler, and their high final results are a consequence of the fact the crawler performs really well in extracting these information from the ACL website.

Chapter 5

Conclusions

In this work, we have presented a literature review on the existing Information Extraction methods and systems; we have introduced IEforNLP, a new system for the extraction of information from scientific articles in Natural Language Processing; and we have generated a relational database with the information extracted.

By specifying our corpus in a restrictive way, we could obtain statistical measures comparable to the ones already mentioned in the state-of-the-art research, as presented in Section 2.3; the evaluation results we obtained have been extensively discussed in Section 4.1. We have also shown that the method generalises fairly, when applied to corpora which are out-of-domain.

The results obtained both with the *in-domain corpus* and the *out-of-domain corpus* show both the qualities and flaws of the system proposed, as can be seen by the reasonable values obtained for the statistical measures computed. The quality of the overall information saved into our relational database evidences the suitability for the extracted information for being used afterwards by decision-makers and/or the research community.

We believe that providing the community with a database of researchers and publications, as we are doing in this work, is highly desirable for future related research. We expect that the historical information that have been put available to the community be useful to identifying opportunities for researchers and institutions and also for foretelling trends in research.

5.1 Future Work

During the design and implementation of IEforNLP, we have identified several possible improvements that could be made, as well as uses for the information extracted by the system. Some of these improvements involve:

- Extract other information from the scientific publications. These might involve the ones already extracted by other IE systems presented in the literature (see the information list in Section 2.3.1).
- Study ways of applying IEforNLP on different high-quality corpora – for example, scientific publications on biomedical and medical fields.

- Analyse the possibility of implementing a hybrid system that would combine the strenght of statistical classifiers with the good results achieved with the use of our rule-based extractors. Attempt at mantaining the high quality results already obtained while also trying to generalise the system application to other domains could be also a highly desirable feature.

- Finally, use the database created with the IEforNLP system in order to build a thorough map of the research in Natural Language Processing world-wide. This is actually one of the main reasons we decided creating IEforNLP for, in the first place. We believe it is highly desirable to have detailed historical information on the research being conducted not only because it might help in the identification of strenghts and possible opportunities for the community, but also because it might provide the (quantitative) grounds for funding bodies and stakeholders to plan and improve research in NLP.

We believe these are highly desirable enhancements for IEforNLP. These enhancements would make the method more robust and allow for better results to be achieved.

Bibliography

[1] ACE. Annotation Guidelines for Entity Detection and Tracking. 2004.

[2] ACL. Association for Computational Linguistics – About the ACL, 2013. Last access on March 14, 2013.

[3] Catriel Beeri, Philip A. Bernstein, and Nathan Goodman. A sophisticate's introduction to database normalization theory. In *Proceedings of the fourth international Conference on Very Large Data Bases - Volume 4*, VLDB '78, pages 113–124. VLDB Endowment, 1978.

[4] Dominique Besagni, Abdel Belaïd, and Nelly Benet. A segmentation method for bibliographic references by contextual tagging of fields. In *Proceedings of the Seventh International Conference on Document Analysis and Recognition - Volume 1*, ICDAR '03, pages 384–388, Washington, DC, USA, 2003. IEEE Computer Society.

[5] Mary Elaine Califf and Raymond J. Mooney. Relational learning of pattern-match rules for information extraction. In *Proceedings of AAAI Spring Symposium on Applying Machine Learning to Discourse Processing*, pages 6–11, Standford, CA, March 1998.

[6] Nancy A. Chinchor. Overview of MUC-7/MET-2. In *Proceedings of the 7th Message Understanding Conference*, 1998.

[7] Nancy A. Chinchor. About MUC/MET, 2001. Last access on May 16, 2012.

[8] K. Bretonnel Cohen and Lawrence Hunter. Getting started in text mining. *PLoS Computational Biology*, 4(1):e20, 01 2008.

[9] David P. A. Corney, Bernard F. Buxton, William B. Langdon, and David T. Jones. Biorat: Extracting biological information from full-length papers. *Bioinformatics*, 20(17):3206–3213, 2004.

[10] G. Doddington, A. Mitchell, M. Przybocki, L. Ramshaw, S. Strassel, and R. Weischedel. The Automatic Content Extraction (ACE) Program–Tasks, Data, and Evaluation. *Proceedings of LREC 2004*, pages 837–840, 2004.

[11] Ronen Feldman and James Sanger. *The Text Mining Handbook – Advanced Approaches in Analysing Unstructured Data*. Cambridge University Press, 2007.

[12] David Fisher, Stephen Soderland, Fangfang Feng, and Wendy Lehnert. Description of the umass system as used for muc-6. In *Proceedings of the 6th Conference on Message understanding*, MUC6 '95, pages 127–140, Stroudsburg, PA, USA, 1995. Association for Computational Linguistics.

[13] Karën Fort, Gilles Adda, and K. Bretonnel Cohen. Amazon mechanical turk: Gold mine or coal mine? *Comput. Linguist.*, 37:413–420, 2011.

[14] D. Freitag and A. McCallum. Information extraction with hmms and shrinkage. In *Papers from the AAAI-99 Workshop on Machine Learning for Information Extraction*, pages 31–36, 1999.

[15] D. Freitag and A. McCallum. Information extraction with hmm structures learned by stochastic optimisation. In *Proceedings of the 17th National Conference on Artificial Intelligence*, pages 584–589, 2000.

[16] Dayne Freitag. Multistrategy learning for information extraction. In *Proceedings of the Fifteenth International Conference on Machine Learning*, pages 161–169. Morgan Kaufmann, 1998.

[17] Hui Han C. Lee Giles, Eren Manavoglu, Hongyuan Zha, Zhenyue Zhang, and Edward A. Fox. Automatic document metadata extraction using support vector machines. In *In JCDL '03: Proceedings of the 3rd ACM/IEEE-CS Joint Conference on Digital Libraries*, pages 37–48, 2003.

[18] Matteo Golfarelli, Dario Maio, and Stefano Rizzi. The dimensional fact model: A conceptual model for data warehouses. *International Journal of Cooperative Information Systems*, 07(02n03):215–247, 1998.

[19] Ralph Grishman and Beth Sundheim. Message understanding conference-6: a brief history. In *COLING '96 Proceedings of the 16th Conference on Computational Linguistics*, volume 1, pages 466–471, Stroudsburg, PA, USA, 1996.

[20] Lynette Hirschman, Jong C. Park, Junichi Tsujii, Limsoon Wong, and Cathy H. Wu. Accomplishments and challenges in literature data mining for biology. *Bioinformatics*, 18(12):1553–1561, 2002.

[21] Jerry R. Hobbs. Information extraction from biomedical text. *J. of Biomedical Informatics*, 35(4):260–264, August 2002.

[22] Jerry R. Hobbs, Douglas Appelt, John Bear, David Israel, Megumi Kameyama, Mark E. Stickel, and Mabry Tyson. Fastus: A cascaded finite-state transducer for extracting information from natural-language text. In Emmanuel Roche and Yves Schabes, editors, *Finite-State Language Processing*, pages 383–406. MIT Press, 1996.

[23] W.J. Hutchins. Alpac: the (in)famous report. *MT News International*, 14:9–12, 1996.

[24] P. S. Jacobs and Lisa F. Rau. SCISOR: extracting information from on-line news. *Commun. ACM*, 33(11):88–97, November 1990.

[25] M.L. Khodra, D.H. Widyantoro, E.A. Aziz, and R.T. Bambang. Information extraction from scientific paper using rhetorical classifier. In *Electrical Engineering and Informatics (ICEEI), 2011 International Conference on*, pages 1–5, July.

[26] Ralph Kimball and Margy Ross. *The Data Warehouse Toolkit: The Complete Guide to Dimensional Modeling (Second Edition)*. Wiley, 2 edition, April 2002.

[27] Nicholas Kushmerick. *Wrapper induction for information extraction*. PhD thesis, 1997. AAI9819266.

[28] John D. Lafferty, Andrew McCallum, and Fernando C. N. Pereira. Conditional Random Fields: Probabilistic Models for Segmenting and Labeling Sequence Data. In *Proceedings of the Eighteenth International Conference on Machine Learning*, ICML '01, pages 282–289, San Francisco, CA, USA, 2001. Morgan Kaufmann Publishers Inc.

[29] A. Lavelli, M. E. Califf, F. Ciravegna, D. Freitag, C. Giuliano, N. Kushmerick, and L. Romano. A critical survey of the methodology for ie evaluation. In *Proceedings of the 4th International Conference on Language Resources and Evaluation*, ELRA, pages 1655–1658, 2004.

[30] Christopher D. Manning and Hinrich Schutze. *Foundations of Statistical Natural Language Processing*. Massachusetts Institute of Technology, 1999.

[31] S. Mukherjea, L.V. Subramaniam, G. Chanda, S. Sankararaman, R. Kothari, V. Batra, D. Bhardwaj, and B. Srivastava. Enhancing a biomedical information extraction system with dictionary mining and context disambiguation. *IBM Journal of Research and Development*, 48(5.6):693–701, Sep.

[32] Fuchun Peng and Andrew McCallum. Accurate Information Extraction from Research Papers using Conditional Random Fields. In *HLT-NAACL04*, pages 329–336, 2004.

[33] Brett Powley and Robert Dale. Evidence-based information extraction for high accuracy citation and author name identification. In *Large Scale Semantic Access to Content (Text, Image, Video, and Sound)*, RIAO '07, pages 618–632, Paris, France, France, 2007. LE CENTRE DE HAUTES ETUDES INTERNATIONALES D'INFORMATIQUE DOCUMENTAIRE.

[34] John D. Prange. Evaluation driven research: The foundation of the tipster text program. In *Proceedings of the TIPSTER Text Program: Phase II*, pages 13–22, Vienna, Virginia, USA, May 1996. Association for Computational Linguistics.

[35] J. R. Quinlan. Learning logical definitions from relations. *Machine Learning*, 5:239–266, 1990.

[36] Lawrence R. Rabiner. A tutorial on hidden markov models and selected applications in speech recognition. In *Proceedings of the IEEE*, pages 257–286, 1989.

[37] Ellen Riloff. Automatically constructing a dictionary for information extraction tasks. In *Proceedings of the eleventh national Conference on Artificial intelligence*, AAAI'93, pages 811–816. AAAI Press, 1993.

[38] Sunita Sarawagi. Information extraction. *Found. Trends databases*, 1(3):261–377, March 2008.

[39] Kristie Seymore, Andrew Mccallum, and Ronald Rosenfeld. Learning hidden markov model structure for information extraction. In *In AAAI 99 Workshop on Machine Learning for Information Extraction*, pages 37–42, 1999.

[40] Amit Singhal. Modern Information Retrieval: A Brief Overview. *Bulletin of the IEEE Computer Society Technical Committee on Data Engineering*, 24(4):35–42, 2001.

[41] S. G. Soderland. Learning text analysis rules for domain-specific natural language processing. Technical report, Amherst, MA, USA, 1996.

[42] Stephen Soderland. Learning information extraction rules for semi-structured and free text. *Machine Learning*, 34:233–272, 1999.

[43] Beth M. Sundheim. Overview of the third message understanding evaluation and conference. In *Proceedings of the 3rd Conference on Message understanding*, MUC3 '91, pages 3–16, Stroudsburg, PA, USA, 1991. Association for Computational Linguistics.

[44] Cynthia A. Thompson, Mary Elaine Califf, and Raymond J. Mooney. Active learning for natural language parsing and information extraction. In *Proceedings of the Sixteenth International Conference on Machine Learning (ICML-99)*, pages 406–414, Bled, Slovenia, June 1999.

[45] K. Zechner. A literature survey on information extraction and text summarization. *Computational Linguistics Program*, 1997.