university of groningen

faculty of mathematics
and natural sciences

Document Understanding for Automatic Proceedings Generation

Rijksuniversiteit, Groningen

Master Thesis

Name:
Jeroen de Groot
S1921320

Supervisor:
prof. dr. ir. M. (Marco) Aiello

Secondary supervisor:
prof. dr. Krzysztof R. Apt

August 16, 2013

Abstract

Conference Management Tools (CMT's) support people involved in running and participating in conferences with process management. Several management tools are available on the World Wide Web, but none of these tools offer a full generation of the proceedings. Together with the fact that automation and digitization of data becomes more and more important we introduce in this thesis a management tool which combines a solution of meta-data extraction and proceedings generation.

Meta-data extraction from research papers is mainly used for indexation of the papers into a digital library. In this thesis, we show that meta-data extraction is also suitable for obtaining correct meta-data which is used for a proper generation of the proceedings for the conference. When meta-data is extracted automatically the user does not have to worry about spelling mistakes which might happen when the data is entered manually, because the extracted data is an exact copy of the data present in the paper. We also show that the automatic extraction improves the usability of the CMT.

For the extraction of the meta-data we applied two different extraction approaches. The title, abstract and index terms are extracted using a rule based approach. For the extraction of the author data we used a machine learning algorithm, in particular a naïve Bayes classifier. The results of those extraction methods are promising. We achieved 99%, 87%, 89% and 96% accuracy for the title, abstract, index terms and authors respectively. This in combination with a low recall (missing results), makes this data very usable for the generation of the proceedings.

Once all the papers are collected for the proceedings and all the meta-data is collected and verified, the proceedings are generated using LaTeX. Based on our findings we conclude that meta-data extraction is suitable in order to improve the usability of the CMT and ensure the meta-data listed in the proceedings is free of spelling errors in at least 95% of the times. The extracted meta-data is also directly usable for indexing of the papers in order to search through them or for distribution.

Keywords: conference management tool, document generation, document understanding, meta-data extraction

Acknowledgement

This thesis has been written for the University of Groningen and CWI Amsterdam. I thank all who helped me in order to achieve this result. Especially I would like to thank Krzysztof R. Apt and Marco Aiello for the coordination and usefull feedback and ideas for the project. I also thank my fellow students and friends for the useful feedback and discussions on this document as well the project itself. I would not have achieved this result without them.

Contents

List of Figures

List of Tables

CHAPTER 1

Introduction

In this thesis a new Conference Management Tool (CMT) is presented with features for information validation and extraction. These features all support the generation of a proper proceedings for the conference. Since users are not consistent in providing meta-data, which is necessary for a well formatted and correct proceedings, all the meta-data will be extracted automatically. This way we acquire consistent meta-data without spelling errors, which might occur when the data is entered manually. By using automatic extraction of the necessary meta-data we also realized a faster paper submission track. In addition to the proceedings generation and meta-data extraction, page numbers can be removed by this CMT, preventing double numbering of the papers once included in the proceedings.

1.1 Conference Management Tools

Managing a conference is a time consuming task. To make this task easier, several conference management tools have been developed. Before conference management tools were introduced, the chair (head of the program committee) of the conference had a hard time managing all the submissions received by mail and later on by e-mail [19]. Nowadays the world wide web is a strong communication medium and perfect for interaction between the chair and all of the author and reviewers within the conference from all over the world [32]. For this reason conference management tools are usually web-based tools. Although these CMT's have a lot in common, they lack in functionalities for the last step in the conference process; the proceedings for the conference including certain contextual checks.

A CMT provides functionalities for authors and reviewers to submit and review papers submitted by the authors who will speak at the conference. The audience of the conference receives all the papers presented during the conference as a bundled package, called the proceedings. Easychair [31] is one of the most popular conference management tools with a user-base of more than 800 thousand users and over 21 thousand conferences. Easychair allows the host of the conference to generate parts of these proceedings automatically. Unfortunately there are no validation rules, like which font types should be used within the submissions. The absence of these rules may result in a corrupt or not properly formatted proceedings when printed.

1.2 Format validation

To achieve a well formatted proceeding, all the submissions included within the proceedings should have a valid format. This means that all the included submissions should use vector based fonts, and should not contain page numbers. All the papers are checked on used font types and based on the font types allowed by the chair the author will receive a notification when incorrect font types are detected. When incorrect fonts are used, the author has to resubmit the paper. It is up to the author to remove the page numbers. When the author does not remove the page numbers, the proceedings manager is able to use the built-in function for page number removal.

Vector based font

Early operating systems relied on bitmap based fonts. When bitmap fonts are scaled they become unreadable, since they are designed for only one display size, see Figure 1.1. Around 1990 Adobe introduced Type 1 fonts [1] based on vector graphics. Unlike the bitmap fonts, the vector based fonts are scalable. Microsoft and Apple developed the their own vector based font called TrueType, so that they did not have to pay royalties to Adobe. These vector based fonts are perfect for printing on every scale without losing reading quality.

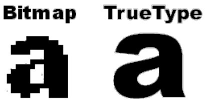


Figure 1.1: Difference between a bitmap and TrueType font.

1.3 Automatic information extraction

Meta-data is necessary for the generation of the proceedings. The title and author data are needed for the index and author information is needed for the author index. Index terms and an abstract are not present in every paper, but this information is useful for review assignment and search queries. The extracted meta-data is also useful for indexing of and searching for papers within the conference or even outside the conference scope. Due to the fact users often provide poor or no meta-data at all, we attempt to extract the following meta-data automatically from the research papers (List 1.2):

- Title
- Abstract
- Index terms (Keywords)
- Authors

List. 1.2: Meta-data, which is extracted (if present) from the research papers submitted to the Easyconf system.

The input files of the Easyconf system are files of the PDF file type. Initially we would like to accept compressed packages with LaTeX files also. This was not feasible due to LaTeX package clashes. The PDF files must contain real text and not a binary image containing the text, since the preprocessor developed for the extraction can not handle such files. OCR (Optical Character Recognition) is required in this case, but this is not in the scope of the project.

Besides the quality improvement of the meta-data by extracting it automatically, the extraction will save the author quite some time submitting a paper with a lot of authors.

Method

Extraction of meta-data from textual documents is mainly done by three approaches [26]. The first approach is the structural approach by looking at font size, text location, string comparison and other quantifiable descriptive measurements. For example the abstract is commonly prefixed by a keyword like 'abstract' and a title is often displayed in the largest font size. The second approach is by web lookup, where mainly the title is extracted and the other relevant data is retrieved from online digital libraries like Google Scholar, Microsoft Academic and IEEE [13][21] The downside of this approach is that not all publications will be retrievable on these sites, since the papers presented on the conference might not be published yet. The last approach involves machine learning algorithms. Machine learning algorithms are able to distinguish between title and not title data by learning from labeled data. For the extraction of the meta-data within the Easyconf system the structural and machine learning approaches are used.

1.4 Research question

Conference management tools make the management process easier for all the parties involved in the conference. There are already quite some tools available like Easychair, OpenConf, ConfTool and EDAS. All these tools have a lot in common, nevertheless none of these tools support a full generation of the proceedings and an automatic meta-data labeling system. In this thesis a new CMT is designed and implemented which has the most common functionalities as described in the comparison of Madhur Jain et al. [14] and has extra functionalities for meta-data labeling, format validation and improvements. Also a full generation of the proceedings is included. The Easyconf system is based on the following research question,

"Will automatic extraction of meta-data and format validation for all submissions to a given conference improve the usability of a conference management tool and ensure a correct generation of the proceedings?"

Before we answer the main research question, the following sub questions are investigated first,

- What is the current state of the art of meta-data extraction from research papers?
- What is the accuracy of the extracted meta-data, in terms of precision and recall?

1.5 Thesis contribution

In this thesis we introduce a CMT which assists the users in the management process of the conference. It helps the authors to submit their work quickly and efficiently by an automatic meta-data labeling system for each submitted paper. Automatic extraction of meta-data reduces spelling errors and increases the usability of the system. For the extraction methods we achieved an extraction accuracy of around 95% with a recall (missing results) of also 95% which ensures us that at least 95% of the data is free of spelling mistakes. Complete proceedings can be generated by the chair or the proceedings manager of the conference, while other CMT's only offer the generation of parts of the proceedings.

In all the related research the meta-data extracted from scientific research papers is used for indexation of the papers into digital libraries. In this thesis we show that the extraction of meta-data increases the usability of paper submission and reduces the amount of incorrect spelled meta-data which is necessary for a properly formatted proceedings.

1.6 Thesis organization

In Chapter 2 we discuss and compare popular conference management tools and their shortcomings. Also the related work for the extraction of meta-data from textual documents is discussed, and how these two are related to each other. Chapter 3 is dedicated to the Easyconf tool itself, explaining all the functionalities within the system and the implementation details for the web-based tool. Chapter 4 presents the methods used for the meta-data labeling system of the submitted papers. We show how the different tags are extracted from the papers and which steps we have taken to improve the extraction results. The results from the extraction methods are shown in Chapter 5. In Chapter 6 we compare the Easyconf system to an existing CMT and answer our research question. We finish this thesis with suggested future work in Chapter 7

CHAPTER 2 Related Work

Easyconf aims at being usable while generating properly formatted proceedings. Automatic extraction of the necessary meta-data needed for the proceedings supports this premise. Together with the generation of a complete proceedings for the conference.

In this chapter we discuss two areas of related work. In Section 2.1 we look at the largest and most widely used conference management tools while the focus lies on the usability and the features supported by the tool. The current state of meta-data extraction and tools for typed documents is discussed in Section 2.2. We take a look at the operation of those tools and the different methods for meta-data extraction and how they may apply in the domain we are working in.

2.1 Management Tools

Conference management tools support the organization in the management part of the conference, i.e. collecting papers from the authors, sending emails, assigning reviewers. With the Internet as one of the most popular interaction platform these days, quite some web-based management tools are developed. Back in 2003 Keita Akagi et al. [3] already published about the feasibility and the implementation model for a paper submission and publication support system. Often speakers on a conference come from various countries thus Internet is a perfect medium to share the work before it will be presented on the conference. Madhur Jain et al. [14] wrote a survey about different existing management tools. They compared the following systems: EDAS, Confious, OpenConf, ConfTool and PaperDyne. These CMT's can be divided in two groups, standalone systems and distributed systems. Standalone systems are systems which must be installed on a server maintained by the organization of the conference itself. Examples of this are the PHP based systems: WCMT, Conftool and OpenConf. Standalone systems can cause installation problems, and it is not possible to continuously integrate new features. Another drawback is that the data is not centralized. You only have access to papers of your own conference, and miss out on the knowledge gathered from other conferences which might cover the same subject material. A distributed system is hosted by the system

CMT	Users / Conference	Open-source	Free	Proceedings
EasyChair	23.500/862.000	no	yes	partly
EDAS	unknown/660	no	no	partly
OpenConf	unknown	yes	yes	no
WCMT	unknown	yes	yes	no
Conftool	unknown	no	no	paper, abstract export
Confious	unknown/10-20	no	yes	no
PaperDyne	unknown	no	yes	no
Microsoft CMT	unknown	no	yes	unknown

Table 2.1: List of commonly used conference management tools.

provider and registration is enough in order to start using the system. All the data is stored by the systems provider. This makes searching through all the data possible. And due to the fact the system is centralized continuous integration of new functionalities is possible. By looking at the restrictions of the standalone systems we choos to design and implement the Easyconf system as a distributed system.

Most of the reviewed CMT's are free of charge. When a user needs extra functionality he has to pay a fee for the EDAS and Conftool systems. The Easyconf system will be free of charge.

Madhur Jain et al. compared the systems based on offered functionality and concluded that EDAS is the richest system in terms of functionality. EDAS has functionality for format checking like margins and two column check, multi-track submissions and so on. The drawback of all those functionality is loss in usability. EDAS feels over engineered, and it is really difficult to keep a clear overview of what is happening within the conference. Systems which are not compared by Madhur Jain et al., but are of interest are EasyChair [31] and CMT from Microsoft. In Table 2.1 we listed the different CMT's.

Paperdyne is not operational anymore, the demo on their website is offline and the last active conference dates back from 2006. When we wanted to evaluate the CMT from Microsoft we were not able to apply for an installation, due to the fact we did not have a conference page as reference. For the OpenConf and WCMT tools it is not possible to identify the user-base, since the tool has to be installed on a machine hosted by the conference itself. EasyChair, EDAS and Microsoft CMT seem to be the most popular CMT's. They offer the most functionalities for the management process, and they show up at different conference pages on the web. None of these tools provide a complete generation of the proceedings, however EDAS and EasyChair allow the user to generate the author index, preface and the index. One of the issues with these systems is that the enclosed papers still have to be numbered manually according to the generated index and author index.

2.2 Meta-data extraction

There is already quite some research done in the field of meta-data extraction from textual documents. For a good generation of the proceedings we need at least a title and the authors listed in the paper. The abstract and index terms can be used for indexing and searching. Before we are able to extract this information, we need to learn about the document layout, and how we need to interpret the different layout features. The documents which are the input for the Easyconf system consist of one specific document class [2], research papers. Therefore we are able to use document specific knowledge and generic knowledge. This knowledge is needed in order to extract different meta-tags. Generic knowledge is the knowledge that titles of the papers are listed in the biggest font. While document specific knowledge may refer that the authors are listed in Helvetica 10pt, which is useful when we compare those pieces of text with the text of the main body in Helvetica 8pt.

Yunhua Hu and Hang Li et al. [12] use machine learning to extract the title from general documents. Giuffrida et al.[9], Zhixin Guo et.al. [10] and Song Mao et al. [20] developed rule-based knowledge systems to extract meta-data. Xiaonan Lu et al. [17] used both. They collected generic and document specific knowledge for the title, which is used in our rule based title extraction method and for the design of the author extraction features. Giuffrida et al. also collected spatial knowledge about the documents. For example, authors are listed immediately under the title in a certain order. This kind of knowledge decreases the amount of data which needs to be analyzed by the different extraction methods. This will give a performance boost in terms of time for the meta-data extraction, and even more interesting we decrease the possibility of misclassifications by reducing the amount of input data.

Otha et al. [23] proposed a method to find an author block within scanned research papers and extract the authors within the block by a specially designed hidden Markov model. The author block is found between the title and the abstract of the document. They achieved a success-rate of 99% in finding the author block and 95% of actually extracting the authors from the block. We adopt the idea of looking for an author block, since it reduces the amount of data to be analyzed by the classifier.

Kazem Taghva et al. [30] mention in their automatic markup system that labeling author data is not possible in most cases by using only context knowledge. For example the author names may be listed in the same font size as the affiliation information in which University names may appear. For this reason person name identification is inevitable. Hui Han, C. Lee Giles et al. [11] proposed a method using Support Vector Machines (SVM) to extract meta-data from research papers.

Hui Han et al. collected information about the domain they applied their meta-data extraction in. For the extraction of the meta-data in the Easyconf system similar knowledge is needed to identify author's. The domain specific information consists of tables with Dutch, Chinese and American first names and surnames collected from the Internet for the domain we are working in.

Existing tools

Beside the methods described in Section 2.2, several tools capable of meta-data extraction are available. Some of those tools are open-source and might be interesting for the Easyconf system. Saleem and Latif [26] tested different existing tools for meta-data extraction from research papers. They combined the results from Mendely[16], Grobid and ParsCit to achieve better meta-data extraction results from research papers. By combining these tools they achieved 95,57% accuracy overall.

Mendeley

Mendeley is a tool for maintaining and ordering research articles. When papers are imported in Mendeley, the relevant meta-data is extracted automatically. These results can be improved by doing a web lookup at Google Scholar. All the information about the imported papers can be exported as XML and may serve as input for the Easyconf system. The downside of Mendeley is that it is only usable with the user interface and does not offer an interface for interaction with our system. Integration of Mendeley within the Easyconf system is not possible by the absence of an interface the system is able to interact with.

PdfMeat

All the meta-data 'extracted' by PdfMeat is actually retrieved by a web-lookup at Google Scholar. Google Scholar has a large database with research papers, only not all papers can be found here. The papers submitted at the Easyconf conferences might not be published yet and therefore not on Google Scholar. This makes PdfMeat not usable for integration with the Easyconf system.

Grobid

Grobid is an open-source Java library for meta-data extraction. The library is big in size and we did not get it working on our machine. We tested the library using a web interface. We achieved similar results as achieved in the paper from Saleem and Latif. The Grobid package is heavy and not easy to integrate with the Easyconf system even if we were able to get it running. We conclude this by the lack of documentation; there is no information available when builds fail or how to interact with the tool.

ParsCit

ParsCit is a Perl solution for labeling meta-data from research papers. ParsCit takes as input a .txt file but does not suggest or support a tool to convert a PDF to txt. Thus we cannot ensure we have a properly formatted input file for the tool. Since the tool cannot exploit context information from the PDF file, the extraction accuracy is low in a lot of cases. We choose not to use ParsCit due to the fact we cannot guarantee a proper input for the tool and the accuracy is low.

CHAPTER 3

Easyconf

Easyconf is a web-based tool allowing the chair of the conference manage the conference from start to end including the generation of a full proceedings for the conference. The focus of Easyconf lies on the automation of the whole process from submission up to and including the generation of the proceedings. The automation of the different parts, such as the extraction of the meta-data from the research papers will increase the usability of the system. It also supports the generation of a properly formatted proceedings because spelling errors are eliminated from the extracted data. In this chapter we describe the different components of the system, together with the implementation details of the system.

3.1 Work-flow

One of the priorities of the system is usability, i.e. the system must be easy in use. The EDAS tool is complicated in its use, due to all the settings that are available in many different views. In order to keep the Easyconf system as simple as possible, a clear and simple work-flow is chosen. All the necessary configurations are made by the completion of the work-flow. This work-flow differs from user to user. For example, the work-flow for an author is much shorter than for the chair, since an author only has to register, select a conference and finally submit his or her work. The chair however, needs to setup the conference and manage all the members and submissions entered into the conference. The work-flow for the author is shown in Figure 3.1

We realize this work-flow will be almost equal in the other conference management tools treated in Chapter 2. But with the automatic meta-data extraction offered by Easyconf, this work-flow is many times faster compared to the other tools. In Chapter 5 there is an in depth comparison of Easyconf against one of the other systems. We also realize that the Easyconf system does not support all the features which are included in the EDAS system. However we believe that we implemented all the functionality that is at least required to manage a conference properly.

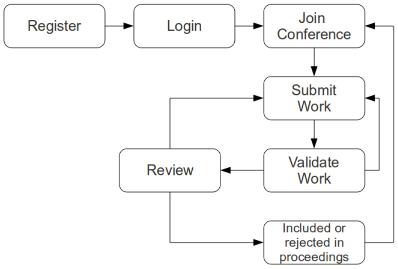


Figure 3.1: Work-flow of the author within the Easyconf system.

3.2 User profiles

Many people are involved during the organization of the conference. These people all have their own role, which varies from the head organizer (the chair) to an author who will present his work on the conference. Every role has its own privileges and restrictions. The Easyconf system supports the following roles;

Chair

The chair has the supervision over the conference, as he created the conference and maintains all the submissions entered into the conference. Therefore he/she is granted all the privileges possible within the system. The chair invites and manages the authors and reviewers for the conference and also sets up the program committee and appoints the proceedings manager.

Program Committee

When a conference hosts many speakers, it is necessary for the chair to have a program committee which helps him with maintaining all the submissions made within the conference. The Program Committee therefore has the same privileges as the chair, except they cannot manage the members within the conference.

Proceedings Manager

When papers are accepted for the conference, they are included in the proceedings for the conference. The Proceedings Manager has access to the proceedings generation process. The Proceedings Manager is also in charge of writing a preface and publishing the proceedings once ready.

Author

When a user joins a conference, he/she has the author and reviewer roles by default. This means they can submit their work to the conference, and edit it afterwards. The privileges of the author are limited to only their own work, although they are able to view work from the other authors within the conference.

Reviewer

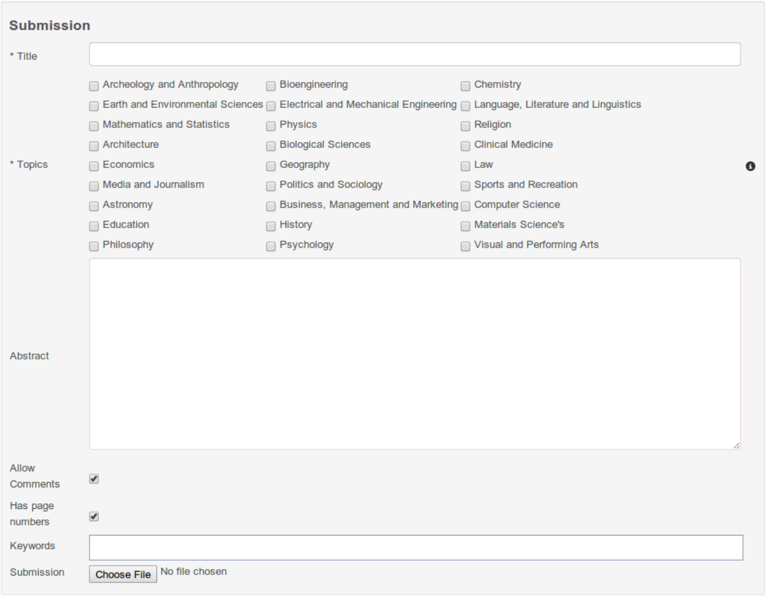
> Reviewers are users who criticize submitted work. Based on those reviews a paper might be edited and resubmitted by the author. A reviewer can view all the submissions made in the proceedings and fill in review forms for them.

When a user needs more privileges, only the chair can set other roles for the user. There is no limitation on the amount of chairs in a conference, since more than one chair might be desirable in big conferences.

When a chair is contributing as an author in a different conference, he can easily switch between conferences by the 'working on' option. When a user is working on a specific conference, all the available roles for the selected conference become active. With this construction just one single account is needed for every user.

3.3 Submission

After a conference is created, authors can join the conference and start submitting their work. The system offers two ways of submitting papers; manual and automatic submission. The submission methods are described in detail in Description 3.3. During the submission phase the paper is checked for the used font types. When these types do not meet the type restrictions set by the chair, an e-mail is sent to the author. Also a reminder is generated in the system and served to the author.



Submission

* Title

* Topics

□ Archeology and Anthropology
□ Earth and Environmental Sciences
□ Mathematics and Statistics
□ Architecture
□ Economics
□ Media and Journalism
□ Astronomy
□ Education
□ Philosophy

□ Bioengineering
□ Electrical and Mechanical Engineering
□ Physics
□ Biological Sciences
□ Geography
□ Politics and Sociology
□ Business, Management and Marketing
□ History
□ Psychology

□ Chemistry
□ Language, Literature and Linguistics
□ Religion
□ Clinical Medicine
□ Law
□ Sports and Recreation
□ Computer Science
□ Materials Science's
□ Visual and Performing Arts

Abstract

Allow Comments ☑

Has page numbers ☑

Keywords

Submission Choose File No file chosen

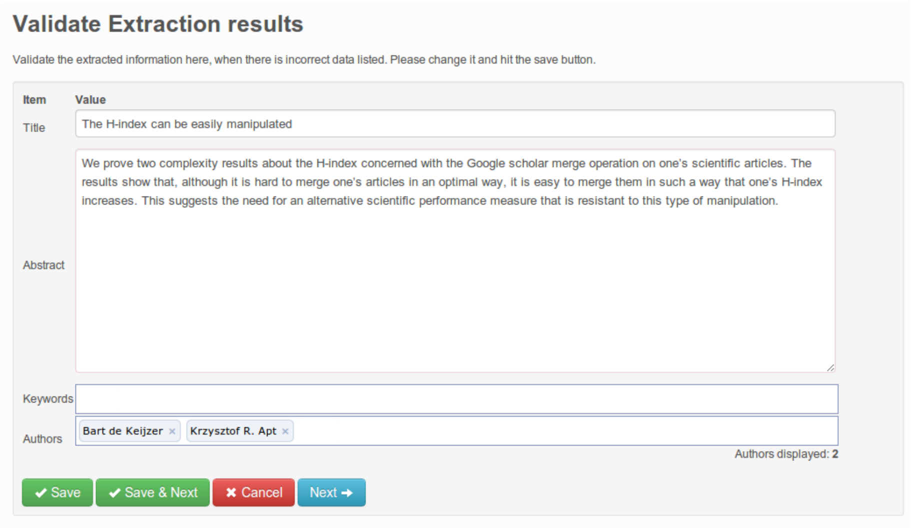
Figure 3.2: Manual submission, where all the necessary field are shown.

Manual submission

Submissions can be done by all users in the conference, regardless of the roles they have. The term manual submission means that the author has to fill in all the relevant data by himself. Figure 3.2 shows the manual submission form and which data has to be provided. This is a time consuming task when a lot of authors contributed in the paper. When many authors have contributed to the paper, a spelling mistake in one of the authors' names is easily made. Most of the data is required for the generation of the proceedings and therefore required to filled in.

Automatic submission

Opposed to the manual submission the author only has to mark the research topics for the submission, since it is not possible to extract those from the research papers at the moment. Once the research paper is submitted, all the relevant meta-data is extracted as described in Chapter 4. The author is asked to validate those results afterwards, since the system cannot guarantee an extraction accuracy of 100%, see Figure 3.3.



Validate Extraction results

Validate the extracted information here, when there is incorrect data listed. Please change it and hit the save button.

Item	Value
Title	The H-index can be easily manipulated
Abstract	We prove two complexity results about the H-index concerned with the Google scholar merge operation on one's scientific articles. The results show that, although it is hard to merge one's articles in an optimal way, it is easy to merge them in such a way that one's H-index increases. This suggests the need for an alternative scientific performance measure that is resistant to this type of manipulation.
Keywords	
Authors	Bart de Keijzer × Krzysztof R. Apt ×

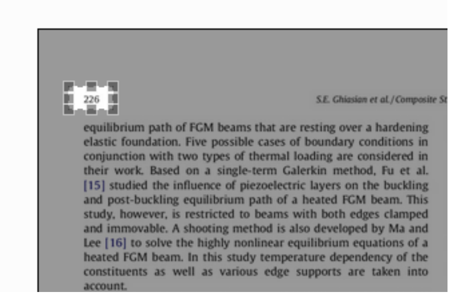
Authors displayed: **2**

Figure 3.3: Validation view of the extracted meta-data.

When an author makes changes to his paper, he can resubmit the paper. All the files which are needed for the automatic extraction of the meta-data are regenerated, because the content of the meta-data might be changed. The meta-data itself is not re-extracted automatically, this might not be desirable when nothing changed regarding the meta-data and the extracted data is already verified. All the versions of the papers are stored within the system, so authors can view and download all the versions they uploaded.

3.4 Page number removal

Within the proceedings all the pages are renumbered, thus a submission should not contain page numbers. If page numbers are present, the author has the option to resubmit the paper without page numbers. The system also offers functionality to remove the page numbers semi-automatically. The author has to mark the location of the page numbers for the even and odd pages, since the page number location may differ for even and odd pages, as can be seen in Figure 3.4. The system places an overlay on those marked areas, so they are not visible in the proceedings anymore.



226 *S.E. Ghiasian et al. / Composite St*

equilibrium path of FGM beams that are resting over a hardening elastic foundation. Five possible cases of boundary conditions in conjunction with two types of thermal loading are considered in their work. Based on a single-term Galerkin method, Fu et al. [15] studied the influence of piezoelectric layers on the buckling and post-buckling equilibrium path of a heated FGM beam. This study, however, is restricted to beams with both edges clamped and immovable. A shooting method is also developed by Ma and Lee [16] to solve the highly nonlinear equilibrium equations of a heated FGM beam. In this study temperature dependency of the constituents as well as various edge supports are taken into account.

Figure 3.4: Selection of an unwanted page number within a submission.

3.5 Review

When an author has submitted his paper, it is accessible for the reviewers within the conference. The reviewers can bid on the paper, which means they indicate that they are interested in actually reviewing the paper. Once the reviewers are assigned to the paper by the chair or program committee, they review the paper by filling in the review form. The authors have the chance to resubmit their work after the reviewing process. During this reviewing process the quality of the paper is determined. When the paper has a certain quality, it may be included in the proceedings and is presented on the conference.

3.6 Proceedings generation

Unlike the conference management tools described in section 2.1 the Easyconf system supports a full generation of the proceedings. Not all papers that are submitted by the authors are included in the proceedings. The quality of the enclosed papers is ensured by having external people, the reviewers, review the papers before they are accepted in the proceedings. Once all the papers are accepted, the proceedings manager creates the proceedings for the conference. The proceedings manager decides the sequence of the paper inclusion, and produces the preface. When the preface is completed and the sequence is final, the proceedings are generated for reviewing. Finally when the proceedings are reviewed and ready for publishing, the proceedings manager publishes the proceedings to the conference page and the authors are notified by e-mail.

The proceedings are generated using LaTeX [25]. LaTeX is a popular document preparation system in many different research sectors. The papers accepted for the proceedings are included using the pfdpages [1] package. This package makes it possible to insert PDF files directly in a LaTeX document. When the proceedings manager has collected all the accepted papers and wrote a preface for the proceedings, all the separate .tex files are generated and compiled with the LaTeX compiler. The proceedings generation process is shown in Figure 3.5

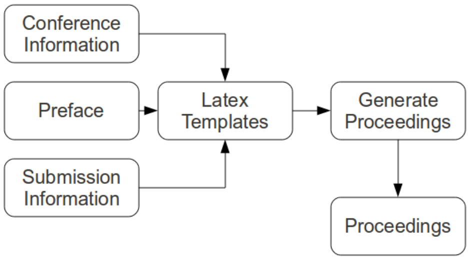


Figure 3.5: Generation process of the conference proceedings

Once the proceedings are compiled the proceedings manager is able to further distribute the compiled PDF file by e-mail and on the conference page within the Easyconf system.

3.7 Implementation

As already mentioned in Chapter 2, conference management tools are easily accessible when realized as a web-application. The Easyconf system is for this reason also designed and implemented as a web-application. In order to keep the system maintainable and modular we choose the Django framework[2] for the implementation together with several design patterns. The most present and important patterns used are the Singleton [8], the Layer pattern [5] and the Model View Controller pattern.

3.7.1 Layer Pattern

The different layers of the Layer Pattern are shown in Figure 3.6.

Presentation Layer

This layer is directly visible to the client (the front-end user). Easyconf

[1] http://www.ctan.org/pkg/pdfpages

[2] https://www.djangoproject.com/

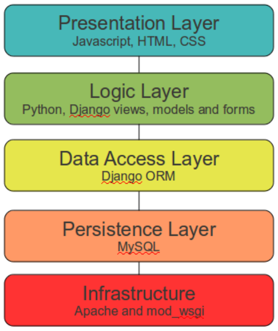


Figure 3.6: The different layers within the Easyconf application.

is a web-based tool and for this reason web and hypertext techniques are used for the presentation layer. The views are rendered in HTML and the design of the content is handled by CSS. For a quick and clean design we used the Twitter Bootstrap[3] framework. Using this framework, we did not have to worry about layout problems across different browsers. On demand data in the presentation layer is retrieved using JavaScript with the JavaScript JQuery library. This on demand data is transfered to the logic layer using JSON objects.

Logic Layer

The logic layer decouples the domain logic from the application. The logic layer prepares the data for the presentation layer. Or domain logic is added to the data when received from the presentation layer and needs to be passed to the data access layer. The logic layer is implemented with Django views, models and forms. Security and data integrity are also handled in this layer. The expected data models, like e-mail or integers are defined in the models and forms. When incorrect data is entered by the user an error is returned to the presentation layer.

Data Access Layer

The data access layer maps the model data from the logic layer to SQL, so it can be inserted or retrieved from the persistence layer. The data mapping is realized by the Django ORM (Object Relational Mapper).

Persistence Layer

The persistence layer consists of a MySQL database. The persistence layer receives the SQL statements from the data access layer and executes them and returns data in case of a select. With the Django ORM model it is possible to switch to another database model, even NoSQL.

Infrastructure

The application is deployed on a Apache web-server with mod_wsgi. mod_wsgi

[3] http://twitter.github.io/bootstrap/

is an Apache module for hosting Python applications which support the Python WSGI (Web Server Gateway Interface) interface.

3.7.2 Singleton Pattern

Training the classifier for the extraction of the author meta-data is a time consuming task. We implemented the Singleton pattern, this means we had to train the classifier just once. Figure 3.7 shows the implementation of the Singleton pattern within the Easyconf system.

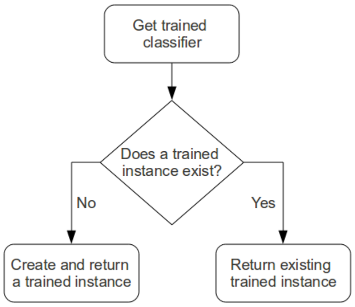

Get trained classifier
Does a trained instance exist?
No
Yes
Create and return a trained instance
Return existing trained instance

Figure 3.7: Singleton implementation of the author extraction classifier

When an author submits a paper the author extraction method requests a classifier instance. When this instance does not exist a new unique instance is trained and returned to the extraction method. By this mechanism we do not have to train the classifier each time a paper is submitted, which makes the extraction much faster.

3.7.3 Model View Controller

Within the Django Framework an object-relational mapper mediates between the data models (defined as Python classes) and the database (Model). A system for processing the requests with a web templating system prepares and renders the data for the presentation layer (View). A regular-expression-based URL dispatcher takes care that the correct view is shown to the end-user (Controller). With this separation of the data models and data handling the system is easily maintainable and modular, since we can easily add or remove views, controllers and data models.

CHAPTER 4

Information extraction

For a proper generation of the proceedings we need meta-data derived from all the papers included within the proceedings. We extract the meta-data listed in List 1.2. The title and the authors form the most important meta-data for a good generation of the proceedings. The abstract and index terms are also extracted for indexing and searching purposes within the Easyconf system. The abstract is also needed when a program containing all the abstracts is created for the conference.

Proved that rule-based extraction methods perform well for the extraction of meta-data from research papers [9]. The title, abstract and index terms are extracted using rule-based pattern matching. The extraction of the authors turned out to be less straightforward and needed a different approach, because author information is not as context rich as the other meta-data. Authors in research papers are not prefixed by consistent keywords like 'written by', 'authors', or are listed in a particular font size. For this reason we use a machine learning algorithm for the extraction of the author data.

The output of the different extraction methods is combined in a meta-data object, which is stored in the database. This object contains all the relevant meta-data for a properly formatted proceedings. The data flow of the extraction model is shown in Figure 4.1

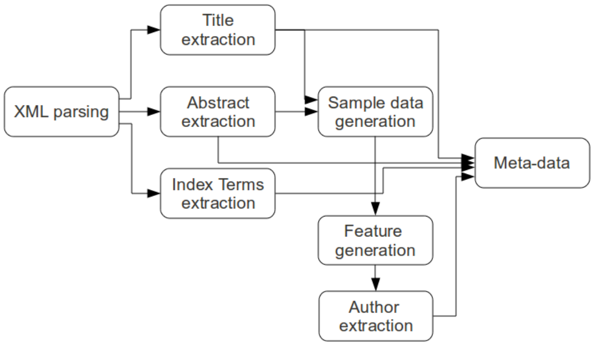


Figure 4.1: Data flow of the extration model of the meta-data extraction

In Section 4.1 we explain in detail what kind of preprocessing is applied in order to prepare the data for the different extraction methods. The rule-based extraction approach for the title, abstract and index terms is presented in Section 4.2 and Section 4.3. The machine learning approach using a naïve Bayes classifier for the extraction of the authors listed in the research papers is described in Section 4.4.

4.1 Prepocessing

The input of the Easyconf system consists of research papers in PDF file format. Before the meta-data is extracted from the PDF files we need to preprocess these files. With this preprocessing phase we prepare the data for the extraction methods and reduce the amount of data [18]. We start reducing the amount of data by separating the first page from the document, because this is were all the data we need is located [9]. Reducing the amount of data indirectly reduces the chance of misclassifications in the extraction methods. It will also speed up the extraction process, because less data needs to be analyzed.

The second step in the preprocessing phase is converting the first page into an XML file. This XML file serves as a direct input for the extraction of the title, abstract and index terms. This XML file is generated using the open-source pdf2xml [6] tool. The pdf2xml tool is a powerful tool capable of converting a PDF file into an XML file containing all the text along with the contextual features like font-size, family, weight etc. The markup of the XML file produced by the tool is shown in Listing 4.1.

```
<page width=”” height=”” number=”” id=””>
    <block id=””>
                <text width=”” height=”” id=”” x=”” y=””>
                        <token id=”” font−name=”” fixed−width=”” bold=”” italic=””
                             font−size=”” font−color=”” rotation=”” angle=”” x=”” y=”
                             ” base=”” width=”” height=””>
                        </token>
                        <token></token>
                        ...
                </text>
                ...
    </block>
</page>
```

Listing 4.1: Markup of the XML file produced by the pdf2xml tool.

Before we are able to use this XML file we need to understand the tags and attributes present in the XML file.

Page

Indicates the page in the range of 1...n where n is the last page of the document.
width: width of the current page in pixels
height: height of the current page in pixels
number: number of the page in the range of 1...n

Block

Represents a single paragraph in the document
id: ID of the paragraph in the form of p1_b1

Text

Represents a single line within a paragraph
width: width of the line in pixels
height: height of the line in pixels
x: x position of the line relative to the document
y: y position of the line relative to the document
id: ID of the line in the form of p1_t1

Token

Represents a single word within a line
width: width of the word in pixels
height: height of the word in pixels
x: x position of the word relative to the document
y: y position of the word relative to the document
base: adjusted y position of the word so that 0,0 is upper left and it is adjusted based on the text direction
angle: rotation angle of the word
rotation: 0 when word is not rotated, 1 otherwise
id: ID of the word in the form of p1_w1
font-size: font-size of the word as a floating point
font-name: name of the used font for the words
font-color: color of the words
bold: yes if word is bold, no otherwise
italic: yes if the word is italic, no otherwise

We choose the XML in Figure 4.1 to be the main input for the meta-data extraction system for the following reasons:

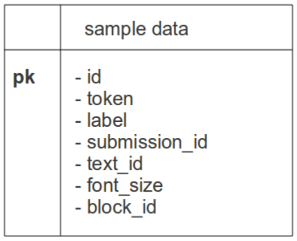
- The produced XML file contains all the text present in the PDF file
- The produced XML file presents logical structure of the PDF file. In each XML the text derived from the PDF is organized in a page, paragraph, line and word hierarchy. This logical structure is useful for the meta-data extraction process. For instance, the abstract is always located in a single paragraph. Thus with the logical structure we easily return the abstract once a paragraph is identified containing the abstract.
- The produced XML file contains information about the format features of every word in the PDF file. From this information we can compute format features for the author extraction. This information is also requisite for the title extraction, since the title is listed in a larger font size than the average font size of the document.

4.1.1 More preprocessing

For the extraction of the authors extra preprocessing is needed. We take the XML file as input for this preprocessing phase. Every text block is divided in sample data blocks which form the input for the author classifier. The tokens of each text block in the XML are separated in one sample data block when:

- A chance in font style (including size, weight and family) occurs
- A comma, the word 'and' or '&' occur

All these sample data blocks are stored in the database along with the following attributes; submission, text block id, font size and block id. When more tokens are present in one block, the mean and median values are calculated over the font-size and also stored in the database. Storing these sample data block makes labeling training data for the classifier a lot easier. The data model for the sample data blocks is shown in Figure 4.2.



sample data

pk

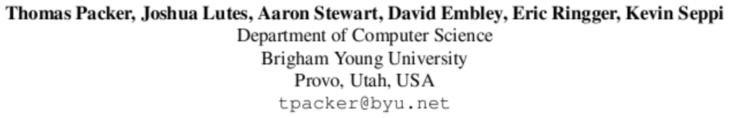
- id
- token
- label
- submission_id
- text_id
- font_size
- block_id

Figure 4.2: Data model for sample data

In this data model, token contains the text of the sample data block. Label is used for training purposes of the classifier and holds one of the values; unknown, other or name. Submission_id is a reference to the submission object and text_id and block_id contain the line and paragraph id derived from the XML file. Finally the mean font size of the token is stored in the font_size field. When we look at Figure 4.3 we achieve the following sample data block from this preprocessing phase (we only list the tokens);

- Thomas Packer
- Joshua Lutes
- Aaron Stewart
- David Embley
- Eric Ringger
- Kevin Seppi
- Department of Computer Science
- Brigham Young University
- Provo

- Utah
- USA
- tpacker@byu.net



Thomas Packer, Joshua Lutes, Aaron Stewart, David Embley, Eric Ringger, Kevin Seppi
Department of Computer Science
Brigham Young University
Provo, Utah, USA
tpacker@byu.net

Figure 4.3: Author listing

This preprocessing phase is executed after the title, abstract and index terms are extracted. By completing the extraction of that data first we have information about their location within the document. With this information we reduce the amount of data which will form the input of the author classifier. We only sample the data between the title and the abstract (or first present paragraph) of the document [23].

4.1.2 Problems

For the second step of the preprocessing we observed problems with old research papers, mainly papers published before 2000. Some of those papers use font types which can not be parsed by the pdf2xml tool. Other old papers are scanned copies and do not contain text at all, since they are actually PDF files containing an image of the scanned document. In these cases the system is unable to convert the PDF file into an XML file. Extraction of the meta-data is impossible to complete. A reminder is generated for the author and the paper is rejected by the system.

Another problem was revealed during the reduction of data for the author classifier. Although the pdf2xml generates a well structured XML it might not have the structure we see in the PDF file. PDF files might store their data in a different order than it is displayed. This means that when the author data is located between the title and the abstract in the PDF file but might be located in the last block of the XML file. In those cases we omit the data reduction and sample the whole page.

4.2 Title

The title of the research paper is needed for a well formatted index within the proceedings. Furthermore it serves as an index while searching for the paper within the system. In Section 4.2.1 we explain the used method for the title extraction, and in Section 4.2.2 we elaborate on the implementation.

4.2.1 Method

The title of the document is extracted using a rule-based pattern matching method. A rule-based system performs well on the papers in the Easyconf system due to the consistent layout. Such a method is easy to implement and does not rely on training data. Only a small set of document specific and generic knowledge is needed. This knowledge forms the rules for the extraction of the title.

Background

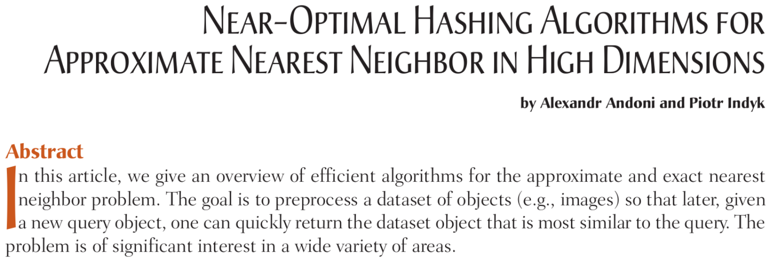
Yunhua Hu and Hang Li et al. [12] used machine learning to extract titles from general documents. Some of the format features they used are interesting for our rule-based system. Like the font size and same paragraph feature. Since a title is assumed to be in the largest font size and might be spread over more text lines. Giuffrida et al.[9] developed a rule based (knowledge-based) system for extraction of meta-data from Post Script files. They made assumptions like "the title is always located in the upper portion of the first page" and "the title is in the largest font size". We adopt those assumptions for our rule based system.

Rules

The following rules are applied in order to extract the title. Where the first two rules are generic rules and the last three are document specific rules, because they apply only on a small subset of all extracted papers.

- Title is on the first page of the document
- Title has largest font size
- Title is not rotated [Exception]
- Title is not equal to the Journal name [Exception]
- Title contains more than one word [Exception]

During the testing phase of our rule-based system we added rules in order to eliminate errors. Since they apply on a subset of the paper we listed them as exceptions rather than rules. arXiv.org places an overlay with an URL referencing to the paper on the first page, in most cases this URL has a larger font size than than the title itself. This URL is located on the left side of the paper and rotated 90°, from this fact we introduced the 'Title is not rotated' exception. This overlay could also be removed by post-processing, but by adding this rule we cover more overlay issues from different publication databases. Elsevier displays the name of the journal in the same font size as the title of the document. Therefore we identify the journal name in Elsevier publications and reject this as the title of the document. By adopting the assumption of Yunhua Hu et al. that a title consists of more than one word we introduced an exception to solve the problem in Figure 4.4, where 'I' is returned as the title.



NEAR-OPTIMAL HASHING ALGORITHMS FOR
APPROXIMATE NEAREST NEIGHBOR IN HIGH DIMENSIONS

by Alexandr Andoni and Piotr Indyk

Abstract

In this article, we give an overview of efficient algorithms for the approximate and exact nearest neighbor problem. The goal is to preprocess a dataset of objects (e.g., images) so that later, given a new query object, one can quickly return the dataset object that is most similar to the query. The problem is of significant interest in a wide variety of areas.

Figure 4.4: When a title with a length of one token is rejected we eliminate that the title extraction function returns 'I' as title.

```
input  : XML of the first page
output: Title
$font\_sizes \leftarrow []$;
$title \leftarrow null$;
foreach $token \in xml.getAllTokens()$ do
    if $token.font\_size \notin font\_sizes$ then
        $font\_sizes.append(token.font\_size)$;
    end
end
foreach $size \in font\_size$ do
    $title \leftarrow xml.getAllTokens(font\_size = size)$;
    if $validate\_title(title)$ then
        return $title$;
    end
end
return $null$;
```

Algorithm 1: Title extraction

4.2.2 Implementation

The title extractor receives an XML file as described in Section 4.1 as input. The different font-sizes occurring in the document are stored in a list, and this list is sorted in decreasing order. For every font size in the sorted list we grab all tokens with that font size. This provides us a list with possible titles. These titles are validated in a separate function where all the exceptions are treated. This way we can easily add more exceptions if needed. The implementation is shown in pseudo-code Algorithm 1.

4.3 Abstract & Index Terms

The abstract can be used when a program containing the abstracts for the conference is desired. While the index terms are used as index values for searching and finding related research papers. The abstract and index terms are not directly needed for a well formatted proceedings. Therefore a lower extraction accuracy will not harm the work-flow and usability of the system. For the extraction of this meta-data we also used a rule-based method as previously described in Section 4.2.1.

4.3.1 Method & Background

The abstract and index terms are extracted by a rule-based method. The used knowledge for the extraction is different from the title extraction. For the extraction of the abstract and index terms we use spatial knowledge. Spatial knowledge refers to the interpretation of certain parts of text when read by humans. This knowledge is used to make documents easier to read, and achieved by using clear paragraph titles for instance. By analyzing research papers we concluded that the abstract and index terms are prefixed with a clear keyword. We derived a list of keywords for the abstract and index terms from the Mendeley system [16]. This list consists of synonyms/variations of the 'abstract' and 'index term' keywords, see Table 4.1. Note that some of these synonyms are actually translations to another language, for example zusammenfassung is German for summary.

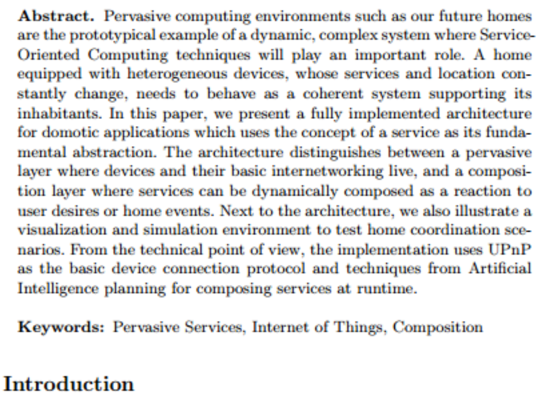
Abstract	**Index Terms**
Abstract	Key-words
Resume	key words
Resumen	Index Terms
Summary	keywords
Synopsis	index-terms
Zusammenfassung	keyword

Table 4.1: Keywords to locate the beginning of the abstract and index terms, derived from the Mendeley system.

With this list of keywords we are able to locate the beginning of the abstract and index terms. Figure 4.5 shows a perfect example of keyword prefixing of the abstract and index terms. We also indicated that the abstract consists of just one paragraph, as the same holds for the index terms. Separating the index terms, in order to present them to the user in a consistent format, requires some extra information. The index terms are separated by a special character or by a line break. We conducted a list of those special characters, so we are able to separate the index terms individually. The index terms are separated by looking for a line break within Elsevier journals.

4.3.2 Implementation

The abstract and index terms extraction algorithms both rely on the XML file produced in the preprocessing step and the set of keywords from Table 4.1. The



Abstract. Pervasive computing environments such as our future homes are the prototypical example of a dynamic, complex system where Service-Oriented Computing techniques will play an important role. A home equipped with heterogeneous devices, whose services and location constantly change, needs to behave as a coherent system supporting its inhabitants. In this paper, we present a fully implemented architecture for domotic applications which uses the concept of a service as its fundamental abstraction. The architecture distinguishes between a pervasive layer where devices and their basic internetworking live, and a composition layer where services can be dynamically composed as a reaction to user desires or home events. Next to the architecture, we also illustrate a visualization and simulation environment to test home coordination scenarios. From the technical point of view, the implementation uses UPnP as the basic device connection protocol and techniques from Artificial Intelligence planning for composing services at runtime.

Keywords: Pervasive Services, Internet of Things, Composition

1 Introduction

Figure 4.5: The beginning of the abstract and index terms are clearly indicated by the 'Abstract' and 'Keywords' keywords.

output for both algorithms is slightly different. The abstract method returns a string containing the abstract, and the index terms method returns a comma separated list containing all the index terms. The algorithm for the extraction of the abstract and index terms is shown in Algorithm 2.

The keyword indicating the abstract or index terms is always the first word of the paragraph. That is why the algorithm only analyzes the first word of the paragraph. Looking for the keywords within the paragraph might result in unexpected output. When the algorithm has found the keyword, it indicates the length of the paragraph. This is necessary, because in some cases the keyword is located in a separate paragraph. When this length is larger than the keyword itself the text of the paragraph is returned, otherwise the text of the current and next paragraph are returned. Preparing the index terms data for presentation purposes is not handled in this algorithm. Finally, the abstract and index terms are stored in a temporary table within the database and stored in the submission table once the author has marked the data as correct.

4.3.3 Problems

The index terms are in all the papers of the validation set prefixed with one of the keywords in Table 4.1. Unfortunately this is not the case for the abstract. Some of the papers in the validation set do have an abstract, but it is just a paragraph without an indicating keyword. Our approach of finding the abstract will fail in those cases. In the case when no abstract is present null is returned, which is an acceptable and expected result.

The extraction accuracy of the index terms suffers from problems earlier indicated of the way data is stored within a PDF file. The index terms are not always placed in a separate block in the XML file, which makes it impossible for the algorithm to locate index terms. We explicitly did not try to solve this by looking for keywords indicating the index terms at the beginning of every line

```
input  : XML file of the first page of the research paper. A list of
         Abstract or Index Terms keywords, keys
output: Abstract or Keywords
$block\_begin \leftarrow null$;
$block\_end \leftarrow null$;
foreach $text\_block \in xml.getAllBlocks()$ do
    if $text\_block.getToken(0) \in keys$ then
        $block\_begin \leftarrow text\_block$;
        if $length(block\_begin.getAllToken()) >$
        $text\_block.getToken(0).length()$ then
            $block\_end \leftarrow block$;
            $break$;
        end
        else
            $block\_end \leftarrow block.next()$;
            $break$;
        end
    end
end
if $block\_begin = null$ then
    return $null$;
end
if $block\_begin = block\_end$ then
    return $block\_begin.text()$;
end
else if $block\_begin \neq block\_end$ then
    return $block\_begin.text().join(block\_end.text())$;
end
```

Algorithm 2: Extraction algorithm for the extraction of the abstract and index terms. Preparing the index terms for returning to the user by converting them in a comma separated list is not included in the algorithm.

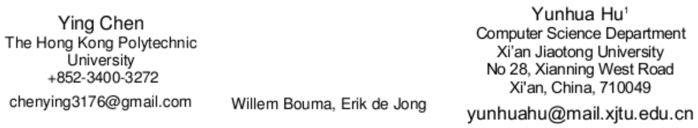
in the document. This is a lot more computational expensive and might result in strange outcomes from the algorithm. In our validation set this problem did not occur often enough to deem it important.

4.4 Authors

The author meta-data is essential for a well formatted proceedings. They are listed in the index as well as in the author-index. And finally they fulfill an index role in the index and searching functionalities within the Easyconf system.

4.4.1 Method

While the research papers use a consistent format for the title and prefix the abstract and index terms with a clear keyword, the authors are listed differently in almost every research paper. From Figure 4.6, it is not directly clear, based on contextual knowledge, which pieces of text we should mark as author data. A rule-based system will not perform well on the variety of research papers we process in the Easyconf system. A machine learing approach is needed to extract the authors with an acceptable accuracy rate.



Ying Chen
The Hong Kong Polytechnic
University
+852-3400-3272
chenying3176@gmail.com

Willem Bouma, Erik de Jong

Yunhua Hu[1]
Computer Science Department
Xi'an Jiaotong University
No 28, Xianning West Road
Xi'an, China, 710049
yunhuahu@mail.xjtu.edu.cn

Figure 4.6: Different types of author presentations in a research paper, where it is not possible to extract the names just on context information as font-size, location in sentence etc.

For this reason we use machine learning for the extraction of the author meta-data, in particular, a naïve Bayes classifier [33]. The naïve Bayes classifier is based on supervised learning. Figure 4.7 illustrates author meta-data extraction based on supervised learning. In the learning and classification process, the sample data blocks generated in the preprocessing phase form the base units. A set of features is extracted over each unit as described in Section 4.4.2. We chose the naïve Bayes classifier for several reasons:

- It needs less training data.
- Training and classification is fast.
- Easy to understand, and minimal tuning is required.

The naïve Bayes classifier is based on the Bayes theorem:

$$p(c_j|d) = \frac{p(d|c_j)p(c_j)}{p(d)}$$

with $p(c_j|d)$ as the probability of instance d being in class c_j, which is what we want to compute for each input. $p(d|c_j)$ is the probability of generating instance d given class c_j, $p(c_j)$ is the probability of occurrence of class c_j and $p(d)$ the probability of instance d occurring.

To simplify the task, the naïve Bayes classifier assumes that the features are independent distributed, thus probability of class c_j generating input d is thereby estimated by:

$$p(d|c_j) = p(d_1|c_j) * p(d_2|c_j) * ... * p(d_n|c_j)$$

where $p(d_1|c_j)$ is the probability of class c_j generating the observed value for feature 1 multiplied by the probability of of class c_j generating the observed value for feature 2 and so on. The features used in our extraction method are explained in the next section.

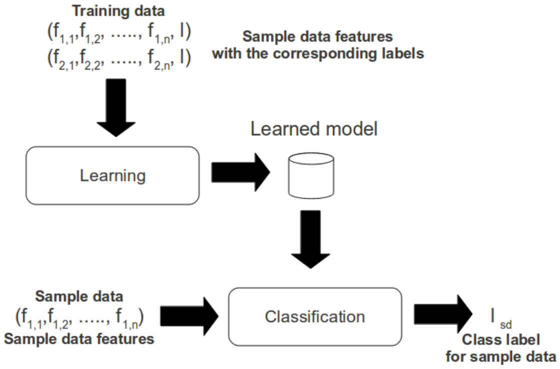


Figure 4.7: Author meta-data extraction based on supervised learning

4.4.2 Feature Extraction

For a good classification we need a good feature selection to distinguish the author information from the other sample data blocks. Three types of features are used; style, font and linguistic/ semantic features. All the features are computed on the sequences outputted by the preprocessor. We believe the sample data blocks are the appropriated units to compute the features on, because during the labeling of the training set each of the authors was represented by a single data block.

Style Features

Style features represent the formatting of the input string.

Capital Letter Mode

This feature represents the usage of capital mode words within the string. The feature is set to one of the following values 0,1, corresponding to non-capital, first/last capital, for the words within the string. Based on the representation of the authors containing of only capital words, Dutch names excepted. For example, the capital letter mode feature is set to 1 for the sequence containing the words 'Marc Romboy', because the first and last word start with a capital. The feature deals correctly with the Dutch prefixes in a name by looking at the capital mode of the first and last word in the string. The capital letter mode is computed by looking at the capital mode of each word in the string. We use wc_0, wc_1 to represent the occurrences of non-capital, first/last capital. The capital mode of the string is represented CM_s and is defined as:

$$CM_s = \begin{cases} 1 & \text{if count}(wc_1) = 2 \\ 0 & else \end{cases}$$

Linguistic and Semantic Features

Liguistic and semantic features represent certain text patterns in the input string.

Person Name

This binary feature indicates if a person name is present in the string. We collected the following sets containing first and surnames from the Internet [15][27][24], see Table 4.2.

First name	Surname	Origin
9 757	9 965	Netherlands
99 725	151 671	America
-	100	China

Table 4.2: Person names distribution collected for the person name feature.

Using common name patterns, this feature is defined as following:

$$PN_s = \begin{cases} 1 & \text{if } w_0 \in f \text{ or } w_n \in s \\ 0 & else \end{cases}$$

Where w_0, w_n are the first and last word in the string, and f, s are the data tables containing the first and surnames.

Negative word

This binary feature represents the presence of a predefined negative word. Most universities are named after a person, for instance T.J. Watson University. By observing the data set we assembled a list of synonyms and variations of the noun; university. With this information the feature is defined as following:

$$NW_s = \begin{cases} 1 & \text{if } w_i \in n \\ 0 & else \end{cases}$$

Where w_i is the ith word in the input string and n the list of negative words.

Numbers

This binary feature represents the presence of a number in the input string. Person names do not contain numbers, but some streets names are named after a person, for instance '23. John Park'. We use n_0 to represent the occurrences of a number in the string. The Number feature is defined as following:

$$N_s = \begin{cases} 1 & \text{if count}(n_0) > 0 \\ 0 & else \end{cases}$$

Word Count

This binary feature represents the length of the string in terms of word occurrences. The sum of the words occurring in the string is calculated by:

$$\sum_{i=1}^{n} w_i$$

where n is the last word in the string. By the observation of the length of a Person Name we set this feature to 0 when the sum of the words in the string is larger than 1 and smaller than 5.

$$WC_s = \begin{cases} 1 & \text{if } 1 < w \leq 4 \\ 0 & else \end{cases}$$

Font Features

Font Size

This feature represents the mean font size of the input string relative to the mean font size of page. As n represents the number of words in the page or input string and f_i the font size of the i$^{\text{th}}$ word, the mean font size is calculated as:

$$MF_{p},_{s} = \sum_{i=1}^{n} f_i \,/\, n$$

Authors are listed in a slightly larger font size than the paragraph text. By this observation the font size feature is defined as following

$$FS_s = \begin{cases} 1 & \text{if } MP_p < MP_s \\ 0 & else \end{cases}$$

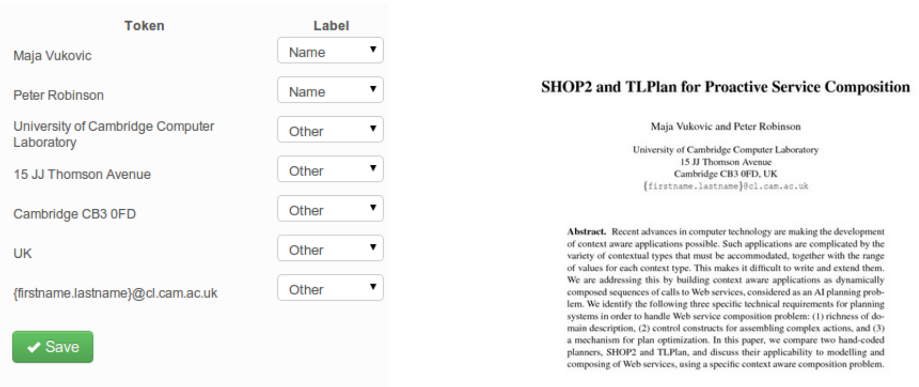
The Capital Letter Mode, Word Count and Font Size features are the leading features, the others are used to increase the precision. Since the naïve Bayes classifier is not sensitive to irrelevant features, we will evaluate some of the features described above in the results in Chapter 5.

4.5 Training

In order to train the classifier approximately 100 papers are labeled manually. Most of those papers are retrieved from public libraries like IEEE, Google Scholar and Elsevier. Also papers from the Student Colloquium course [29] are included. These are not available on any of the online databases and are therefor interesting for meta-data extraction. All the training data is stored in the database which is also used and accessed by the Easyconf system. This makes it is easy to adjust the size of the training and testing sets, and new labeled data can easily be added to the training set.

Labeling

The first step of the labeling process is that the papers are handled by the preprocessor. The output is presented in a view to the user, which makes the labeling of the sample data quick and easy, as can be seen in Figure 4.8. We only had to indicate every token string if it is a name or not. We use 3 types of labels; 'unknown', 'name' and 'other'. When a token is generated by the preprocessor it has the unknown label by default. The other two labels are used to train the classifier to match unknown input. We located all the papers which are used for the training of the classifier in a seperate conference, this made it very easy to adjust the size of the training set an we have a clear overview of the training set.



Token	Label
Maja Vukovic	Name ▾
Peter Robinson	Name ▾
University of Cambridge Computer Laboratory	Other ▾
15 JJ Thomson Avenue	Other ▾
Cambridge CB3 0FD	Other ▾
UK	Other ▾
{firstname.lastname}@cl.cam.ac.uk	Other ▾

SHOP2 and TLPlan for Proactive Service Composition

Maja Vukovic and Peter Robinson

University of Cambridge Computer Laboratory
15 JJ Thomson Avenue
Cambridge CB3 0FD, UK
{firstname.lastname}@cl.cam.ac.uk

Abstract. Recent advances in computer technology are making the development of context aware applications possible. Such applications are complicated by the variety of contextual types that must be accommodated, together with the range of values for each context type. This makes it difficult to write and extend them. We are addressing this by building context aware applications as dynamically composed sequences of calls to Web services, considered as an AI planning problem. We identify the following three specific technical requirements for planning systems in order to handle Web service composition problem: (1) richness of domain description, (2) control constructs for assembling complex actions, and (3) a mechanism for plan optimization. In this paper, we compare two hand-coded planners, SHOP2 and TLPlan, and discuss their applicability to modelling and composing of Web services, using a specific context aware composition problem.

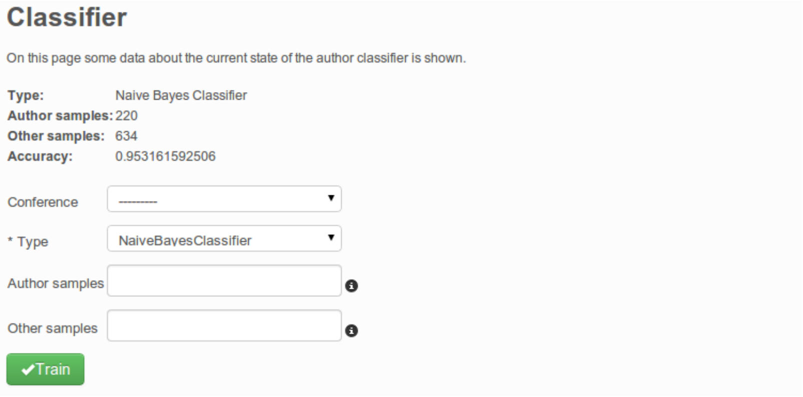
Figure 4.8: A view included in the Easyconf system for sample data labeling. This data is used for validation and training data for the classifier.

4.6 Implementation

Our author meta-data extraction system is implemented using Python and a naïve Bayes classifier from the Natural Language Toolkit NLTK [4]. Although the naïve Bayes classifier is fast in training and classification of unlabeled data. The training process is a time consuming task. With a training set of about 1000 samples it takes more than 20 sec to train the classifier. The system this was tested on was a:

- HP 6730b laptop
- Intel T9700 processor at 2.53GHz
- 4GB DDR2 800MHz RAM
- 7200 RPM HDD
- Ubuntu 10.04 LTS

Training the classifier every time a user submits a paper would make the whole process slow and unproductive. Therefore we used the Singleton pattern [8] for the classifier as described in Section 3.7. We also provided a view within the Easyconf system that allows the user to retrain the classifier with a specific amount of training samples, as can be seen in Figure 4.9.



Classifier

On this page some data about the current state of the author classifier is shown.

Type: Naive Bayes Classifier
Author samples: 220
Other samples: 634
Accuracy: 0.953161592506

Conference ---------
* Type NaiveBayesClassifier

Figure 4.9: View for adjusting specific settings for the training of the author classifier

CHAPTER 5

Results

We have shown that the Easyconf system is able to support the chair in the process of managing a conference from the start up to and including the generation of the proceedings. The main focus of the system is usability, and for this reason most of the processes are automated. In this chapter we evaluate the system in its usability and we show results from the extraction methods used for the meta-data extraction.

5.1 Trial setup

For the trial we asked 3 users to use the Easyconf and EasyChair system and fulfill the following tasks.

- Register to the system.
- Create a conference.
- Submit 5 pre-selected papers, and use the quick submission within the Easyconf system.
- Generate proceedings, or parts of the proceedings in case of the EasyChair system.

During this trial they had to measure the time spent for each task and finally answer some questions about the user experience. The different users are familiar with the different phases of a conference, and fully understand the definitions like chair, submission, proceedings etc. None of these users had ever worked with the Easyconf or EasyChair system before or used another similar conference management tool.

Evaluating the system by a small amount of people gives us insights about the effectiveness of the automatic extraction of the meta-data and the automatic generation of a full proceedings. Although we cannot rely on such a small test group to provide definitive answers about the usability, we feel that it gives an insight on the current level of usability of the compared systems in this trial. The set of used papers for this trial is chosen for the variety in contributing authors and the presence of an abstract or index terms. The set consists of papers with foreign authors, which might increase the misspelled author rate when the data is added manually (EasyChair).

Directly at the beginning of the system evaluation a problem occured. Creating a conference on the EasyChair system is done by a request and the user has to wait for acceptance. This process can take up to a few days. Thus we had to skip this part and remove all the data for the conference in EasyChair for each subject.

The time spent completing the tasks gives us a indication of the usability. The measured time to complete the tasks is shown in Table 5.1.

User	**EasyChair**	**Easyconf**	**Decrease**
Gerhard	32 minutes	12 minutes	20 minutes
Thomas	41 minutes	16 minutes	25 minutes
Steven	36 minutes	14 minutes	22 minutes

Table 5.1: Execution time for submitting five papers, and generating proceedings or parts of it.

When we look at the consumed time in Table 5.1 we see a significant difference in the time spent submitting papers. This is partly related to the automatic meta-data extraction and partly to the more user-friendly user interface of the Easyconf system. The users were able to find the proceedings functionalities faster within the Easyconf system. Finally, checking if the data was extracted correctly took even less time than checking for spelling mistakes made in the EasyChair system.

5.1.1 Opinion

After the users completed the different tasks, we asked them a few open questions to get their opinion about the system. With these questions we are able to get an more clearly insight of the usability of the system.

As an author you usually submit just one paper at a conference management system. And the meta-data extraction has to deal with errors which has to be corrected by the user. Which submission process do you prefer, automatic or manual, and why?
Gerhard: Automatic, since this is less prone for errors. Just checking the values for correctness is much easier, simpler and faster. It will save a lot of time, and make it more accessible for adding papers. When typing in the data manually, or even copy/pasting them, it leaves to much room for error. Especially if is foreign to you. And if names for instance are incorrectly spelled, it does not come across as professional.
Thomas: Automatic, I really liked the fact all the authors were listed directly afte the submission and only needed a quick review before saving. Even copying the authors from the paper and pasting them into the EasyChair system was more time consuming.

Steven: Automatic, it saved me a lot of time. Especially with the foreign names, I had to check them several times for spelling errors after I had entered them into the EasyChair system.

Would you recommend the Easyconf system to conference chair's due to the user friendly submission system and the complete proceedings generation?
Gerhard: Yes, I would. The EasyChair tool took me a lot more time to enter all the data. The speed and user friendly interface of Easyconf makes it a great tool for this purpose.
Thomas: Yes, the Easyconf system was easier to understand and I liked the automatic extraction of all the necessary information.
Steven: Yes, especially for a small conference the automatic generation of the proceedings is a nice feature. The automatic information extraction is not really necessary, but its a cool feature, I liked it.

The system has a lower precision than recall rate, which means it returns more authors than listed in many cases and almost never misses one. Do you find this very disturbing?
Gerhard: I only experienced this once, and the recognition of it being a false positive and deletion of it was easy. I was not hindered by it what so ever. When entering data manually, it takes more time to type and check your spelling.
Thomas: It was pretty easy to recognize the wrong result, therefor it was not disturbing for me. It was also easy to remove the wrong results.
Steven: It did not disturb me at all, the system made it really easy to remove the incorrect listed authors.

Are there things that definitely needs to be improved for either of the management tools used in this test?
Gerhard: It was easy to use, and it felt complete. I see no necessary improvements at this time.
Thomas: I haven't used the system long enough to give an answer to this question at the moment. In general I liked the Easyconf system more than the EasyChair system, because it was easier in use and it has a nice and fresh user interface.
Steven: As I mentioned before, I liked the proceedings, but the layout of the proceedings is a bit boring. A variety of layouts would be a nice addition.

5.1.2 Evaluation

In order to evaluate the usability of the system we use some of the usability meassueres from Ahmed Seffah et al.[28].

Effectiveness

Effectiveness indicates if the software is capable to enable the user to *achieve specified tasks with accuracy and completeness.*

Within the Easyconf system the user is forced to follow a specific work-flow when he starts submitting his work. The user is also provided with hyperlinks to all kind of functionality and breadcrumbs so he can easily navigate through the system at any point. With this work-flow and hyperlinks with a clear declaration about their functionality, we enable the user to complete the tasks with accuracy and completeness.

Productivity

Productivity indicates the level of effectiveness achieved in relation to effort consumed by the user.

Automatic extraction of meta-data decreases the effort consumed by the author significantly. Together with the decrease of spelling errors in the submission meta-data needed for the proceedings generation, the Easyconf system has a high productivity.

Satisfaction

Satisfaction is related to the subjective response from the users.

We estimate the satisfaction based on the answers in Section 5.1.1. Based on those opinions the Easyconf has a good level of satisfaction. The users liked the simple and clean design together with the automatic extraction method which really improved the productivity of the system. One of the users also mentioned the ease of creating proceedings automatically.

Learnability

Learnability indicates the ease in which features required for achieving particular goals can be mastered

Due to the clear declarations of all the hyperlinks in the system with pop-overs with additional information the learnability of the Easyconf system is good. The test subjects in the trial indicated that the Easyconf system was easier to understand, because in the other system they were not all able to find the proceedings generation functionalities without our support.

Although we only evaluated the system with a small group of users, we believe we achieved a good usability level for the system. The meta-data extraction reduces the rate of spelling mistakes made by the authors and also increases the productivity and level of satisfaction. With the automatic generation of the proceedings, anyone can hold a conference and does not need any knowledge of document creation in order to create proceedings for the conference.

5.2 Extraction

For the extraction of the meta-data we used a rule-based approach for the title, abstract and index terms, which is based on proven research. For the author extraction we used, opposed to the related work a naïve Bayes classifier instead of a Support Vector Machine or HMM. The use of the naïve Bayes classifier made the implementation much easier and we booked good results with it when compared to the related work.

5.2.1 Dataset

Research papers show slightly different layouts in different research areas. The layout might also differ a little between publishers. The Easyconf system supports paper submission from different research areas and publishers. Therefore a set of 103 research papers from 2 different research areas and from several publishers are used as input for the validation of the extraction methods. The research areas were Computing Science and Medicine. Medicine papers often have more contributing authors than the Computing Science papers. The Computing Science set contains of papers from the student colloquium course [29], publications by M. Aiello and K.R. Apt and randomly selected papers from IEEE and Elsevier Journals. The Medicine papers are randomly selected from PubMed, which is a digital library of Medicine [7]. All the papers are written in English.

The papers in the validation set are processed by the system and manually checked, with the view as can be seen in Figure 3.3. This adjusted data is considered as the correct data. We need this data in order to validate the results from the different meta-data extraction methods. The validation set is located in a separate conference, so we have a clear insight over the validation set. The list of papers used for the train and test set can be found in the Appendix A.

We assume that every research paper has at least a title and a contributing author. Not all the papers used in the verification dataset contain index terms or an abstract. In Table 5.2 the coverage for the different meta-tags is listed.

Tag	Coverage
Title	103
Abstract	101
Index Terms	65
Authors	103

Table 5.2: Coverage of the different meta-data in the validation set

5.2.2 Evaluation measures

The results of the extraction methods are represented in terms of precision and recall [22]. Where tp (true positive) indicate correctly labeled meta-data and fp (false positive) are incorrectly labeled meta-data. fn (false negative) indicates meta-data which is not labled as meta-data, thus missing results. Finally tn (true negative) is data which does not fall in any of the meta-data categories and has correctly not received an label by any of the extraction methods. The evaluation measures are defined as follows:

Precision: $P = \frac{tp}{tp+fp}$
Recall: $R = \frac{tp}{tp+fn}$
F-measure: $F = 2 \cdot \frac{precision \cdot recall}{precision+recall}$
Accuracy: $A = \frac{tp+tn}{tp+tn+fp+fn}$

Precision represents the relation between the amount of meta-data requested and the amount of meta-data retrieved by the system. Recall represents the relation between the amount of meta-data retrieved by the system and the amount of meta-data which was expected. The F-measure is the weighted average of those measurements. Finally the accuracy represent the proportion of true results (both true positive and true negative) in the whole dataset. With these measures we have a good insight of the performance of the extraction methods. From these measures it is directly clear if we have a lot absent results for instance. These measures are also used in the related work, which makes the comparison a lot easier.

5.2.3 Experimental Results

For the author extraction validation we used 4-fold cross validation. The results for the authors shown in Table 5.3, Table 5.4 and Figure 5.1 are those averaged over 4 trials. The results for the other meta-tags are not changing when the number of runs increases, because static sets of rules are defined.

Tag	Precision	Recall	F-meassure	Accuracy
Title	99.03%	100.00%	99.51%	99.03%
Abstract	93.62%	92.63%	93.12%	87.13%
Index Terms	96.05%	92.41%	94.19%	89.02%
Authors	83.81%	95.51%	89.28%	96.96%

Table 5.3: Results of the meta-data extraction over the validation set.

All the meta-data is individually matched with the validation set. When titles are not equal they are marked as incorrect. We extracted only one title incorrectly. The recall for the title will always be 1.0, since the algorithm always returns a title, because it returns the piece of text with the largest font

Tag	Correct	Incorrect	Absent	Total	Coverage
Title	102	1	0	103	103
Abstract	88	6	7	101	101
Index Terms	292	12	24	316	65
Authors	383	74	18	401	103

Table 5.4: Results of the extraction of the meta-data in terms of numbers.

size, which is not rejected by any of the rules. For the abstract we compute a similarity ratio. With this ratio we eliminate cases that the abstract is indicated as incorrect when small differences like spaces are present in the validation set. This ratio is computed by the Ratcliff & Obershelp algorithm:

Ratio: $r = 2 * \frac{M}{T}$

In which M represents the matching characters and T is the combined length of both strings. When this ratio is smaller than 0.95 the abstract is marked as incorrect. The authors and index terms are matched the same way as the title. When an index term or author name is equal to the value in the labeled validation set they are marked as correct.

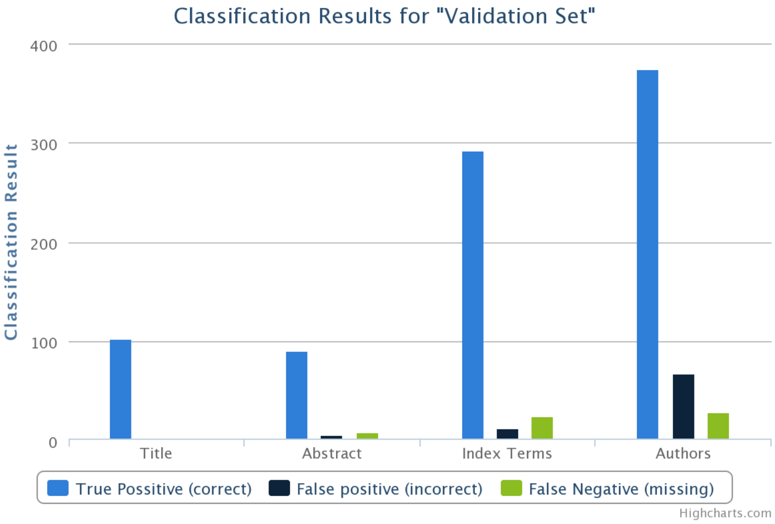


Figure 5.1: Results of the meta data extraction

5.2.4 Person Name Feature

The person name feature produces heavy database interaction, since 2 queries are executed for each input to the classifier. In order to analyze the impact of the person name feature we conducted two different types of classification, with and without the person name feature enabled. It turns out that this feature has a low impact on the recall, but has a high impact on the precision of the classification. Currently the system asks the author for a quick review on the extracted meta-data. But in a ideal situation we want the system to extract the necessary data without further need for manual validation, thus we also want a precision as high as possible. A significant improvement of the precision is desirable, because it reduces the chance of unwanted meta-data in the final proceedings. In order to increase the precision over time, all unknown first and surnames are stored in the database after the validation process. The results are shown in Table 5.5.

Name features	Precision	Recall
Off	73.72%	95.81%
On	83.81%	95.51%

Table 5.5: Results of the author extraction with and without the named features enabled

5.2.5 Evaluation

The results we booked with the extraction methods are promising. The author extraction performs well related to the SVM implementations used in the related research. No impending problems occurred during the extraction of the title. Due to the consistent format of the title it was a trivial task to design rules the extraction of the title. The extraction of the abstract performed a bit less, because of the fact that not all the abstracts are prefixed with a keyword. In this case, our algorithm is unable to extract them. The extraction of the index terms suffered from the fact the data in a PDF is not always stored in the way it is presented to the user. A workaround for this problem would be checking for the occurrence of one of the indicating keywords at the beginning of every sentence.

The misclassifications of the authors is related to the fact that strings like 'Denton Texas' has a similar feature vector as 'John Snow' since they have the same values for the capital, person name and word count feature. Eliminating these misclassifications is not a trivial task. Authors are not always listed on the same line. This makes the introduction of line specific features not possible. With line specific features we could eliminate misclassifications like 'Denton Texas'.

Nevertheless we are satisfied with the low recall, because we made it very easy for the user to remove incorrect results. Having a low recall ensures that no spelling mistakes are made in ∼95% of the author data. The title data is for the same reason free of spelling errors in about ∼99% of all titles present in the Easyconf system. We are very pleased with these results, because the title together with the author data are the most important for a well formatted proceedings.

In order to get a good overview of the results we booked from the meta-data extraction we compared our results against the related work in Table 5.6.

Work	**Method**	**Title**	**Abstract**	**Index**	**Authors**
Ours	Rule,nBayes	99.03%	87.13%	89.02%	96.96%
M. Ohta et al. [23]	HMM	-	-	-	95.42%
Z. Guo et al.[10]	Rule	90.50%	86.37%	-	93.33%
S. Mao et al. [20]	Rule	96.36%	98.18%	-	89.09%
H. Han et al.[11]	HMM	98.30%	98.40%	98.50%	93.20%
H. Han et al.[11]	SVM	98.90%	97.50%	99.20%	99.30%

Table 5.6: Related extraction results in terms of accuracy.

The values displayed in Table 5.6 are accuracy measures, because most of the related work gave no insights of the precision and recall. From this table we may conclude we can be satisfied with the results we booked from the extraction methods. Our rule-based title extraction outperforms all the related work. By looking at the other rule-based results, we have to improve the methods for the extraction of the abstract and index terms. Finally extraction of the authors is only outperformed by the SVM method from Hui Han et al.

CHAPTER 6

Discussion and Conclusion

We successfully developed a conference management tool with a high usability level, where the automatic extraction of the meta-data supported a well formatted generation of the conference proceedings. The different methods for the meta-data extraction produce satisfying results in terms of precision and recall. It also increased the usability of the system, because authors can submit their work more quickly.

6.1 Trial

In the previous chapter we listed all the results derived from the trial. Although the trial was done by a small amount of people, it gave us an indication about the usability of the system. The automatic extraction of meta-data was received positively. Submitting papers became childs play and it was not longer necessary to check for spelling mistakes after filling in the forms for the relevant meta-data. One of the subject liked the automatic generation of the proceedings, but did not like the style of it. The other users did not place any comments on the layout of the proceedings. In contrast to the EasyChair system we also add page numbers to all the submissions included in the proceedings. When a conference wants a custom cover, they can still use the generated proceedings, and only need to replace the cover.

6.2 Extraction

In the related work the extracted meta-data is used for index values of papers in digital libraries. Misclassified data will be on influence on the search results in those cases. However, when a part of the data is extracted correctly, searching on several keywords will still return valid results. The impact of misclassified data within the Easyconf system is much higher. Misclassified meta-data will result in a proceedings with unwanted content. For this reason, the system will ask the author for a brief review in order to correct the misclassifications. Due to the low recall of the extraction methods, the meta-data extraction does support a well formatted proceedings. The meta-data is present in ∼95% of the times, when looking at the title and author data. We only look at the title and author extraction results, in order to answer the research question, because they are required for a proper generation of the proceedings. Based on the output results of the title and author methods, we are sure no spelling mistakes are made in that part of the meta-data, since this data is a direct copy of the data present in the submissions. To kill two birds with one stone, this meta-data could be used for indexing purposes simultaneously.

6.3 Proceedings

The automatic extraction of the meta-data supports a generation of well formatted proceedings. The generated proceedings are ready for use, since all the accepted submissions are included and numbered according to the index. The author index is numbered according to order the papers are included in the proceedings, which makes searching through the document easier. At the moment the produced proceedings are ready for publishing to all the authors within the conference. In the future more layout options for the proceedings will be a nice addition.

6.4 Final Thoughts

By looking at the research question and the results we booked with the extraction methods, we believe we are able to support a well formated generation of the proceedings by using automatic extraction of the meta-data. We achieve a good overall accuracy related to the title and author extraction. Together with the low recall in the author extraction we guarantee that at least 95% of the author data is free of spelling errors and about 99% of the title data is free of spelling errors. Concluding from the results of the usability evaluation the automatic meta-data extraction also improved the usability of a conference management system. Users get their work done faster and do not have to worry about spelling errors. The validation step right after the extraction is inevitable, since the system is unable to guarantee a extraction accuracy of 100%. The users in the trial were not impeded by it at all. And with this validation step we ensure meta-data with a very low error-rate. Making the data also suitable for indexation of the submitted papers in other systems.

CHAPTER 7

Future work

The future work for the Easyconf system could be divided into two areas, the tool itself and the meta-data extraction part.

7.1 Easyconf

The composition for the proceedings is static at the moment. In the future it will be a good addition if the proceedings manager would be able to modify the layout for the proceedings. This would make the system more attractive in use, since you don't need an external tool anymore in order to make the proceedings match the conference style.

7.2 Meta-data Extraction

With the rule based extraction methods we achieved acceptable extraction accuracy. The only problem with the rule based system is that it is not as generic as a machine learning approach. The rule based system is highly context dependent. In the scope of the Easyconf system the context dependency is not a problem at the moment, but when we want to accept general documents to the system we need another approach.

The integration of all meta-data extraction in just one approach is better for the maintainability of the system. With additional features the naïve Bayes classifier will be able to extract the title, abstract and index terms from the research papers.

APPENDIX A

Papers used for training & validation

This appendix contains all the papers used for the training set for the author classifier. Also all the papers used for the validation of the meta-data extraction methods are listed.

A.1 Training

1. Sequential Projection Learning for Hashing with Compact Codes
2. Survey of Conference Management Systems
3. A Survey of Peer-to-Peer Storage Techniques for Distributed File Systems
4. An Unsupervised Method for Author Extraction from Web Pages Containing User-Generated Content
5. Lexical Post-Processing Optimization for Handwritten Word Recognition
6. Conditional Restricted Boltzmann Machines for Structured Output Prediction
7. Survey of SIFT Compression Schemes
8. Domain-Speciic Keyphrase Extraction
9. Tahoe The Least-Authority Filesystem
10. Efcient Graph-Based Image Segmentation
11. Automatic Document Metadata Extraction using Support Vector Machines
12. Proper Name Extraction from Non-Journalistic Texts
13. Metadata Extraction from Chinese Research Papers Based on
14. Scalable Similarity Search with Optimized Kernel Hashing
15. Hashing with Graphs
16. A Multiphase Level Set Framework for Image Segmentation Using the Mumford and Shah Model
17. High-Dimensional Visualizations
18. Extracting Person Names from Diverse and Noisy OCR Text
19. Information Extraction from Research Papers by Data Integration and Data Validation from Multiple Header Extraction Sources
20. Image Segmentation in Video Sequences:
21. Automatic Extraction of Table Metadata from Digital Documents
22. Context Thesaurus for the Extraction of Metadata from Medical Research Papers
23. FASTUS: A Finite-state Processor for Information Extraction from Real-world Text
24. Growth of multiwall carbon nanotubes in an inductively coupled plasma reactor

25. Supervised Hashing with Kernels
26. Spectral Hashing
27. A Fast Multilevel Implementation of Recursive Spectral Bisection for Partitioning Unstructured Problems
28. Modeling the Author Bias Between Two On-line Computer Science Citation Databases
29. Title Extraction from Bodies of HTML Documents and its Application to Web Page Retrieval
30. Composite Hashing with Multiple Information Sources
31. Minimal Loss Hashing for Compact Binary Codes
32. Focused Named Entity Recognition Using Machine Learning
33. SSVM : A Simple SVM Algorithm
34. SPEC Hashing: Similarity Preserving algorithm for Entropy-based Coding
35. Twosh: A 128-Bit Block Cipher
36. Complementary Hashing for Approximate Nearest Neighbor Search
37. Sloppy hashing and self-organizing clusters
38. Contour and Texture Analysis for Image Segmentation
39. Self-Taught Hashing for Fast Similarity Search
40. Biblio: automatic meta-data extraction
41. Towards Automatic Real Time Preparation of On-Line Video Proceedings for Conference Talks and Presentations
42. Extracting Author Meta-Data from Web using Visual Features
43. Extracting Person Names from Diverse and Noisy OCR Text
44. Automatic Extraction of Titles from General Documents using Machine Learning
45. Locality-Sensitive Binary Codes from Shift-Invariant Kernels
46. Random Maximum Margin Hashing
47. Automatic Metadata Extraction and Classification of Spreadsheet Documents Based on Layout Similarity
48. The Automatic Content Extraction (ACE) Program Tasks, Data, and Evaluation
49. The WHIRLPOOL Hashing Function
50. Iterative Quantization: A Procrustean Approach to Learning Binary Codes for Large-scale Image Retrieval
51. Defeating Vanish with Low-Cost Sybil Attacks Against Large DHTs
52. Knowledge-Based Extraction of Named Entities
53. Spherical Hashing
54. Learning to Hash with Binary Reconstructive Embeddings
55. Using Multiple Segmentations to Discover Objects and their Extent in Image Collections
56. Laplacian Co-hashing of Terms and Documents
57. Packing bag-of-features
58. Image Segmentation by Data-Driven Markov Chain Monte Carlo
59. Design and Implementation of a Collaborative Conference Management System
60. An empirical study of the naive Bayes classier
61. Chord: A Scalable Peer-to-peer Lookup Service for Internet Applications
62. A Dynamic Feature Generation System for Automated Metadata Extraction in Preservation of Digital Materials
63. Authors Names Extraction from Scanned Documents
64. The Research and Implementation of Turning Conference Management System into a Service
65. GROBID: Combining Automatic Bibliographic Data Recognition and Term Extraction For Scholarship Publications

66. Hybrid Image Segmentation Using Watersheds and Fast Region Merging
67. BIOINFORMATICS ORIGINAL PAPER
68. PolyUHK: A Robust Information Extraction System for Web Personal Names
69. Accurate Information Extraction from Research Papers using Conditional Random Fields
70. Graph Cuts and Efcient N-D Image Segmentation
71. An Evaluation of an Automatic Markup System
72. Information Extraction
73. Automatically Constructing a Dictionary for Information Extraction Tasks
74. Learning to Search Efciently in High Dimensions
75. Crawling BitTorrent DHTs for Fun and Prot
76. eBizSearch: A Niche Search Engine for e-Business
77. Attribute Discovery via Predictable Discriminative Binary Codes
78. Abstract Recommendation with Assistance of Interactive User Prole Extraction
79. OceanStore: An Architecture for Global-Scale Persistent Storage
80. Data Extraction and Label Assignment for Web Databases
81. A Performance Evaluation and Examination of Open-Source Erasure Coding Libraries For Storage
82. Accelerating Error Correction in High-Throughput Short-Read DNA Sequencing Data with CUDA
83. Kernelized Locality-Sensitive Hashing for Scalable Image Search
84. A Multiscale Random Field Model for Bayesian Image Segmentation
85. RIPEMD-160: A Strengthened Version of RIPEMD
86. Large-Scale Image Retrieval with Compressed Fisher Vectors
87. Fast and Inexpensive Color Image Segmentation for Interactive Robots
88. Learning to Hash Logistic Regression for Fast 3D Scan Point Classication
89. Fast Author Name Disambiguation in CiteSeer
90. LDAHash: Improved matching with smaller descriptors
91. Compact Hashing with Joint Optimization of Search Accuracy and Time
92. Reference Metadata Extraction from Scientific Papers
93. Intelligent Content Based Title and Author Name Extraction from Formatted Documents

A.2 Validation

1. The H-index can be easily manipulated
2. Performance of Hashing-Based Schemes for Internet Load Balancing
3. Web Service Composition - Current Solutions and Open Problems
4. Logic for physical space
5. Robust Audio Hashing for Content Identification
6. Deletion of the NMDA-NR1 Receptor Subunit Gene in the Mouse Nucleus Accumbens Attenuates Apomorphine- Induced Dopamine D1 Receptor Trafcking and Acoustic Startle Behavior
7. Interoperation, Composition and Simulation of Services at Home
8. Continual Planning with Sensing for Web Service Composition
9. Fibrogenesis in Crohns Disease
10. GoalMorph: Partial Goal Satisfaction for Flexible Service Composition
11. Amygdala Receptors Modulate Delayed Downregulation of Dopamine Activity following Restraint
12. Towards Semantic Services for Sensor-Rich Information Systems
13. Validation of murine dextran sulfate sodium-induced colitis using four therapeutic agents for human inflammatory bowel disease
14. Alterations of the mucosal immune system in inflammatory bowel disease
15. QuickCheck: A Lightweight Tool for Random Testing of Haskell Programs
16. Towards Physical Mashups in the Web of Things
17. Textual Article Clustering in Newspaper Pages
18. Role of interleukin-21 isoform in dextran sulfate sodium (DSS)-induced colitis
19. Hyperactive hypothalamus, motivated and non-distractible chronic overeating in ADAR2 transgenic mice
20. Spectral Hashing
21. An orally active matrix metalloproteinase inhibitor, ONO-4817, reduces dextran sulfate sodium-induced colitis in mice
22. Evaluation of fibrosis in precision-cut tissue slices
23. Declarative Enhancement Framework for Business Processes
24. Peritoneal brosis is mouse strain dependent
25. Intestinal brosis in inammatory bowel disease: progress in basic and clinical science
26. Locality-Sensitive Hashing Scheme Based on p-Stable Distributions
27. The Power Grid as a Complex Network: a Survey
28. Behaviour of Crohns disease according to the Vienna classication: changing pattern over the course of the disease
29. From model cell line to in vivo gene expression: disease-related intestinal gene expression in IBD
30. Mineralocorticoid receptor throughout the vessel: a key to vascular dysfunction in obesity
31. Similarity Search in High Dimensions via Hashing
32. Chameleon Hashing and Signatures
33. Common Knowledge in Email Exchanges

34. A study of the possible association between adenosine A receptor gene polymorphisms and attention-decit hyperactivity disorder traits
35. ONTOLOGY-SUPPORTED AUTOMATIC SERVICE CHAINING FOR GEOSPATIAL KNOWLEDGE DISCOVERY
36. A Semantics-based Middleware for Utilizing Heterogeneous Sensor Networks
37. Social Network Games
38. Social Networks with Competing Products
39. Towards Decentralized Trading: A Topological Investigation of the Dutch Medium and Low Voltage Grids
40. Undominated Groves Mechanisms
41. ROBUST IMAGE HASHING
42. SOCRADES: A Web Service based Shop Floor Integration Infrastructure
43. Intestinal Mesenchymal Cells
44. SHOP2 and TLPlan for Proactive Service Composition
45. Semantic-Based Planning of Process Models
46. Social Network Games with Obligatory Product Selection
47. Paradoxes in Social Networks with Multiple Products
48. Loss of transforming growth factor signalling in the intestine contributes to tissue injury in inammatory bowel disease
49. Synthy: A System for End to End Composition of Web Services
50. Physical activity and environmental enrichment regulate the generation of neural precursors in the adult mouse substantia nigra in a dopamine-dependent manner
51. An Observational Study of the Association between Adenovirus 36 Antibody Status and Weight Loss among Youth
52. Green Cloud Computing: Balancing Energy in Processing, Storage, and Transport
53. Wireless Sensor Networks Using Android Virtual Devices and Near Field Communication Peer-To-Peer Emulation
54. CAKE Classifying, Associating & Knowledge DiscovEry An Approach for Distributed Data Mining (DDM) Using PArallel Data Mining Agents (PADMAs)
55. A Tutorial on Hidden Markov Models and Selected Applications in Speech Recognition
56. DroidMat: Android Malware Detection through Manifest and API Calls Tracing
57. Integrated Resource Management in Cognitive Radio
58. Collaboration-Based Cloud Computing Security Management Framework
59. Predicting Radio Resource Availability in Cognitive Radio - an Experimental Examination
60. QoE-Driven Channel Allocation Schemes for Multimedia Transmission of Priority-Based Secondary Users over Cognitive Radio Networks
61. Enhancing Stealthiness & Efficiency of Android Trojans and Defense Possibilities (EnSEAD)
62. Cloud Computing Learning
63. The Issues of Cloud Computing Security in High-speed Railway
64. On the Automatic Categorisation of Android Applications
65. Research on Cloud Computing Data Security Model Based on Multi-dimension
66. Spectrum-Aware Dynamic Channel Assignment in Cognitive Radio Networks
67. Security Threats in Cloud Computing

68. Impact of Security Risks on Cloud Computing Adoption
69. ANDROID PRIVACY
70. Realization of Open Cloud Computing Federation Based on Mobile Agent
71. Cloud Computing Security Threats and Responses
72. CDA: A Cloud Dependability Analysis Framework for Characterizing System Dependability in Cloud Computing Infrastructures
73. On the Throughput and Spectrum Sensing Enhancement of Opportunistic Spectrum Access Cognitive Radio Networks
74. Quality Data for Data Mining and Data Mining for Quality Data: A Constraint Based Approach in XML
75. Research on Data Mining Models for the Internet of Things
76. Ad-Hoc Association-Rule Mining within the Data Warehouse
77. Selfish Attacks and Detection in Cognitive Radio Ad-Hoc Networks
78. Composition Motions of Android
79. Bottom-Up Generalization: A Data Mining Solution to Privacy Protection
80. Postbuckling analysis of variable angle tow plates using differential quadrature method
81. A new approach to three-dimensional exact solutions for functionally graded piezoelectric laminated plates
82. An experimental study of creep behavior of lightweight natural fiber-reinforced polymer composite/honeycomb core sandwich panels
83. Free vibration analysis and optimization of composite lattice truss core sandwich beams with interval parameters
84. Nonlinear transient response of fibre metal laminated shallow spherical shells with interfacial damage under unsteady temperature fields
85. Damage resistance and damage tolerance of dispersed CFRP laminates: The bending stiffness effect
86. Thermo-electro-mechanical vibration of piezoelectric nanoplates based on the nonlocal theory
87. Defects in composite structures: Its effects and prediction methods A comprehensive review
88. A nonlinear cohesive model for mixed-mode delamination of composite laminates
89. Linear statics and free vibration sensitivity analysis of the composite sandwich plates based on a layerwise/solid-element method
90. Highly accurate nonlinear three-dimensional finite element elasticity approach for biaxial buckling of rectangular anisotropic FGM plates with general orthotropy directions
91. Nonlinear free vibration of laminated composite rectangular plates with curvilinear fibers
92. Effect of mass diffusion on the damping ratio in a functionally graded micro-beam
93. An investigation of Mode I and Mode II fracture toughness enhancement using aligned carbon nanotubes forests at the crack interface
94. Free vibration analysis of functionally graded carbon nanotube-reinforced composite plates using the element-free kp-Ritz method in thermal environment
95. Honeycomb composites with auxetic out-of-plane characteristics
96. Damage resistance and damage tolerance of dispersed CFRP laminates: Effect of ply clustering
97. An exact solution for the free vibration analysis of laminated composite cylindrical shells with general elastic boundary conditions

98. A multiaxial fatigue damage model for fibre reinforced polymer composites
99. Instability of eccentrically stiffened functionally graded truncated conical shells under mechanical loads
100. Hybrid laminated-glass plate: Design and assessment
101. Effective topologies for vibration damping inserts in honeycomb structures
102. Dynamic buckling of suddenly heated or compressed FGM beams resting on nonlinear elastic foundation
103. EastWest gradient in the incidence of inflammatory bowel disease in Europe: the ECCO-EpiCom inception cohort

APPENDIX B

Latex template

This appendix contains the templates we used for the generation of the proceedings. The content listed within this templates is rendered using the render_to_template function included in the Django framework. With this render function we are able to directly access the objects from the Easyconf system. For example we are able to render the name of the conference on the cover page by using {{conference.name}}, see Listing B.1.

```
\newcommand*{\titleGM}{\begingroup
\hbox{
  \hspace*{0.2\textwidth}
  \rule{1pt}{\textheight}
  \hspace*{0.05\textwidth}
  \parbox[b]{0.75\textwidth}{

    {\noindent\Huge\bfseries {{conference.name}} }\\[2\baselineskip]
    {\large \textit{ {{conference.importantdate_set.all.0.start_date}} }}\\[4\baselineskip]
    {\Large \textsc{ {{conference.city}}, {{conference.get_country_display}} } }
    \vspace{0.5\textheight}
  }
}
\endgroup}


\titleGM
```

Listing B.1: LATEX template for the cover page of the proceedings.

```
\twocolumn
\sloppy
\chapter*{Foreword}

{{preface}}

\onecolumn
```

Listing B.2: LATEX template of the foreword of the proceedings.

```
\documentclass[oneside]{book}
\usepackage[final]{pdfpages}
\usepackage[left=1.2in, right=1.0in, top=1.0in, bottom=1.0in]{geometry}
\usepackage{makeidx}
\usepackage{graphicx}
\usepackage{titlesec}
\makeindex
\usepackage[nottoc]{tocbibind}
\renewcommand{\indexname}{Author Index}

\titleformat{\chapter}[display]
    {\normalfont\Large\raggedleft}
    {\MakeUppercase{\chaptertitlename}%
    \rlap{ \resizebox{!}{1.2cm}{\thechapter} \rule{15cm}{1.2cm} } }
    {10pt}{\Huge}
\titlespacing∗{\chapter}{0pt}{30pt}{20pt}

\begin{document}
    \thispagestyle{empty}
    \include{cover−page}
    \pagestyle{plain}
    \setcounter{page}{1}
    \include{info−page}

    \frontmatter
        \include{foreword}
        \tableofcontents

    \mainmatter
        \setcounter{page}{3}
        {% for paper in submissions %}
            {% for author in paper.authors.all %}
                \index{ {{author.last_name}}, {{author.first_name}} }
            {% endfor %}

            \includepdf[fitpaper=true,
                        pages=−, pagecommand={\thispagestyle{plain}},
                        addtotoc={1,
                                  chapter,
                                  {{forloop.counter}},
                                  { {{paper.title}}, {% for author in paper.authors.all %}
                                      {% if forloop.last %} \textnormal{ {{author.get_full
                                      _name}} } {% else %} \textnormal{ {{author.get_
                                      full_name}}, } {% endif %}{% endfor %} },
                                  { {{paper.title}} } }] {”{{path}}/{{paper.pk}}/{{paper.
                                      file.noextension}}”}
        {% endfor %}

        \restoregeometry

    \backmatter
        \printindex
\end{document}
```

Listing B.3: LaTeX template for the main page which is needed for the generation of the proceedings.

Bibliography

[1] Adobe. Adobe type 1 font format. `http://partners.adobe.com/public/developer/en/font/T1_SPEC.PDF`, 1990.

[2] Marco Aiello, Christof Monz, Leon Todoran, and Marcel Worring. Document understanding for a broad class of documents. *International Journal on Document Analysis and Recognition*, 5(1):1–16, 2002.

[3] Keita Akagi, Tatsuya Furukawa, and Masashi Ohchi. Feasibility implementation of paper submission and publication support system over the internet. In *SICE 2003 Annual Conference*, volume 2, pages 1681–1686 Vol.2, 2003.

[4] Steven Bird, Edward Loper, and Ewan Klein. *Natural Language Processing with Python.* OReilly Media Inc., 2009.

[5] Frank Buschmann, Regine Meunier, Hans Rohnert, Peter Sommerlad, and Michael Stal. *Pattern-Oriented Software Architecture, Volume 1: A System of Patterns.* Wiley, Chichester, UK, 1996.

[6] Herve Dejean and Emmanuel Giguet. pdf2xml [computer software]. `http://sourceforge.net/projects/pdf2xml/`, 2007-2012.

[7] National Center for Biotechnology Information. Us national library of medicine. `http://www.ncbi.nlm.nih.gov/pubmed`, 2013. [Online; accessed June-2013].

[8] Martin Fowler. *Patterns of Enterprise Application Architecture.* Addison-Wesley Longman Publishing Co., Inc., Boston, MA, USA, 2002.

[9] Giovanni Giuffrida, Eddie C. Shek, and Jihoon Yang. Knowledge-based metadata extraction from postscript files. In *In Proceedings of the Fifth ACM Conference on Digital Libraries*, pages 77–84. ACM Press, 2000.

[10] Zhixin Guo and Hai Jin. A rule-based framework of metadata extraction from scientific papers. In *International Symposium on Distributed Computing and Applications to Business, Engeneering and Science*, 10th, 2011.

[11] Hui Han, C. Lee Giles, Eren Manavoglu, Hongyuan Zha, Zhenyue Zhang, and Edward A. Fox. Automatic document metadata extraction using support vector machines. pages 37–48, 2003.

[12] Yunhua Hu, Hang Li, Yunbo Cao, Li Teng, Dmitriy Meyerzon, and Qinghua Zheng. Automatic extraction of titles from general documents using machine learning. *Information Processing & Management*, 42(5):1276 – 1293, 2006.

[13] IEEE. Ieee xplore. `http://http://ieeexplore.ieee.org/`, 2013. [Online; accessed 25-April-2013].

[14] Madhur Jain, Tribhuwan K. Tewari, and Sandeep K. Singh. Survey of conference management systems. In *International Journal of Computer Applications*, volume 2, May 2010.

[15] KNAW. Netwerk naamkunde. `http://www.naamkunde.net/`, 1986-2011. [Online; accessed May 2013].

[16] Mendeley Ltd. Mendeley [computer software]. `http://www.mendeley.com/`.

[17] Xiaonan Lu, Brewster Kahle, James Z. Wang, and C. Lee Giles. A metadata generation system for scanned scientific volumes. In *Proceedings of the 8th ACM/IEEE-CS joint conference on Digital libraries*, JCDL '08, pages 167–176, New York, NY, USA, 2008. ACM.

[18] Jasdeep Singh Malik, Prachi Goyal, and Akhilesh K Sharma. A comprehensive approach towards data preprocessing techniques & association rules.

[19] Aleksander Malinowski and Bogdan Wilamowski. Paper collection and evaluation through the internet. In *Industrial Electronics Society, 2001. IECON '01. The 27th Annual Conference of the IEEE*, volume 3, pages 1868–1873 vol.3, 2001.

[20] Song Mao, Jong Woo Kim, and George R. Thoma. A dynamic feature generation system for automated metadata extraction in preservation of digital materials. In *Proceedings of the First International Workshop on Document Image Analysis for Libraries (DIAL'04)*, DIAL '04, pages 225–, Washington, DC, USA, 2004. IEEE Computer Society.

[21] Microsoft. Microsoft academic search. `http://academic.research.microsoft.com/`, 2013. [Online; accessed 25-April-2013].

[22] David L Olson and Dursun Delen. *Advanced Data Mining Techniques*. Springer, 2008. page 138.

[23] Manabu Otha, Shun Yamasaki, Takayuki Yakushi, and Atsuhiro Takasu. Authors' names extraction from scanned documents. In *Digital Information Management, 2007. ICDIM '07. 2nd International Conference on*, volume 1, pages 67–72, 2007.

[24] John Pasden. The top 100 chinese surnames. `http://www.sinosplice.com/learn-chinese/chinese-vocabulary-lists/the-top-100-chinese-surnames`, 2013. [Online; accessed May 2013].

[25] Latex Project. Latex. `http://www.latex-project.org/`, 2012. [Online; accessed March-2013].

[26] Ozair Saleem and Seemab Latif. Information extraction from research papers by data integration and data validation from multiple header extraction sources. In *Proceedings of the World Congress on Engineering and Computer Science*, volume 1, 2012.

[27] Social Security. Popular baby names, 2013. [Online; accessed May 2013].

[28] Ahmed Seffah, Mohammad Donyaee, Rex B. Kline, and Harkirat K. Padda. Usability measurement and metrics: A consolidated model. *Software Quality Control*, 14(2):159–178, June 2006.

[29] Rein Smedinga, Michael Biehl, and Femke Kramer, editors. *Proceedings 9th Student Colloquium 2011-2012*. University of Groningen, Bibliotheek der R.U., 2012.

[30] Kazem Taghva, Allen Condit, and Julie Borsack. An evaluation of an automatic markup system. Technical report, In: Proceedings of the IS&T/SPIE 1995 International Symposium on Electronic Imaging Science and Technology, 1995.

[31] Andrei Voronkov. Easychair conference system. `http://www.easychair.org/easychair.cgi`, 2004. [Online; accessed 26-March-2013].

[32] Yuchi Xuebiao, Lee Xiaodong, Jin Jian, and Yan Baoping. Measuring internet growth from dns observations. In *Future Information Technology and Management Engineering, 2009. FITME '09. Second International Conference on*, pages 420–423, 2009.

[33] Harry Zhang. The optimality of naive bayes. In Valerie Barr and Zdravko Markov, editors, *Proceedings of the Seventeenth International Florida Artificial Intelligence Research Society Conference (FLAIRS 2004)*. AAAI Press, 2004.