Erasmus University Rotterdam

IBEB – B.Sc. Thesis

Stanisław Guner – 371788

Disambiguation of Scientific References in a Patent Database:

A Project to Facilitate Economic Research and Policy Evaluation

Supervisor: Emiel Caron, PhD[1]

Co-reader: Bauke Visser, PhD

Date: 30 July 2015

Version: Final

[1] Due to the collaborative nature of this project it must be acknowledged that the authorship of the work presented in this paper belongs also to the supervisor.

ABSTRACT

PATSTAT database stores information on patent applications and publications. One of its tables, with the code name TLS214, stores information about scientific references, that are cited by patents. In the 2014 version of PATSTAT this table holds almost 24 million citations. As such, this table is a potentially powerful resource to investigate the relation between science and technology. However, TLS214 is poorly designed and it proves problematic for researchers and policy makers to use the information it contains. The project, which is described in this paper, presents an automated record disambiguation procedure, that aims to provide a reliable way for the scientific community to verify hypothesis on the TLS214 table. To this end, we employ basic, string cleaning methods alongside some pattern harmonization techniques. Next, we extract bibliographic information and use it to detect pairs of records that are potential duplicates. A pair scoring system is used to reject certain pairs and the final clusters of duplicates are obtained with use of a clustering algorithm.

ACKNOWLEDGEMENTS

I would like to sincerely thank my supervisor – Prof. Emiel Caron of the ESE for his extraordinary help and cooperation on this project. Also, I would like to thank Jos Winnink of CWTS for devoting his time to talk with us about issues of patent statistics and for supplying us with a dataset, that is used for evaluation of the performance of our method.

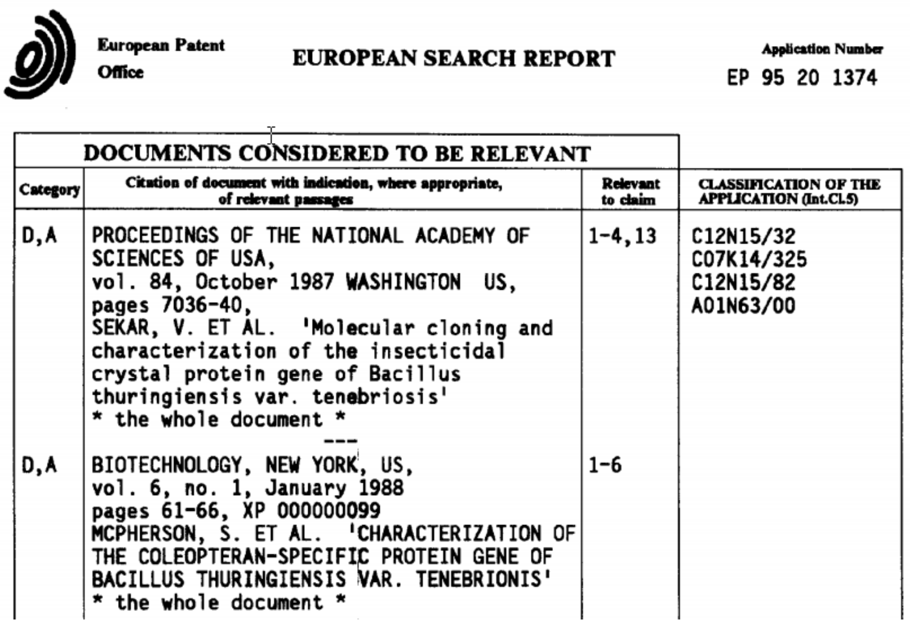
Table of Contents:

1 INTRODUCTION

1.1 Background

1.1.1 Scientometrics

Scientometrics is a science that provides (quantitative) measures for evaluation of scientific output through analysis of bibliographic information (Leydesdorff & Milojević, 2015). It is most commonly used in studies on impact and reach of scientific works. In economics its most prominent use can be found in policy evaluation and research on innovation. Such studies make use of the fact that bibliographic data is often linked to other economic phenomena. One example of such a relation is a citation of a scientific publication in a patent publication. Figure 1-A shows an excerpt from a patent publication[2]:



European Patent Office

EUROPEAN SEARCH REPORT

Application Number
EP 95 20 1374

DOCUMENTS CONSIDERED TO BE RELEVANT

Category	Citation of document with indication, where appropriate, of relevant passages	Relevant to claim	CLASSIFICATION OF THE APPLICATION (Int.Cl.5)
D,A	PROCEEDINGS OF THE NATIONAL ACADEMY OF SCIENCES OF USA, vol. 84, October 1987 WASHINGTON US, pages 7036-40, SEKAR, V. ET AL. 'Molecular cloning and characterization of the insecticidal crystal protein gene of Bacillus thuringiensis var. tenebriosis' * the whole document * ---	1-4,13	C12N15/32 C07K14/325 C12N15/82 A01N63/00
D,A	BIOTECHNOLOGY, NEW YORK, US, vol. 6, no. 1, January 1988 pages 61-66, XP 000000099 MCPHERSON, S. ET AL. 'CHARACTERIZATION OF THE COLEOPTERAN-SPECIFIC PROTEIN GENE OF BACILLUS THURINGIENSIS VAR. TENEBRIONIS' * the whole document *	1-6	

Figure 1-A Scientific citations in a patent publication.

Science, technology and economy are intrinsically connected with each other. Understanding how undeniably constitutes valuable economic knowledge. However, in order to understand the connections between those phenomena a suitable methodological environment needs to be created (in the like of the *wheel of science*[3]), that will support formulation of new hypothesis and their swift verification on real life data.

[2] Publication number EP0682115A1. Can be conveniently viewed at Google Scholar: https://www.google.nl/patents/EP0682115A1?cl=en&dq=EP0682115A1&hl=en&sa=X&ved=0CCAQ6AEwAGoVChMI0YiQtbL7xgIVhL1yCh3CzAjQ

[3] 1. Theory 2. Hypothesis 3. Empirical Verification 4. Empirical Generalization

1.1.2 PATSTAT AND TLS214 Table

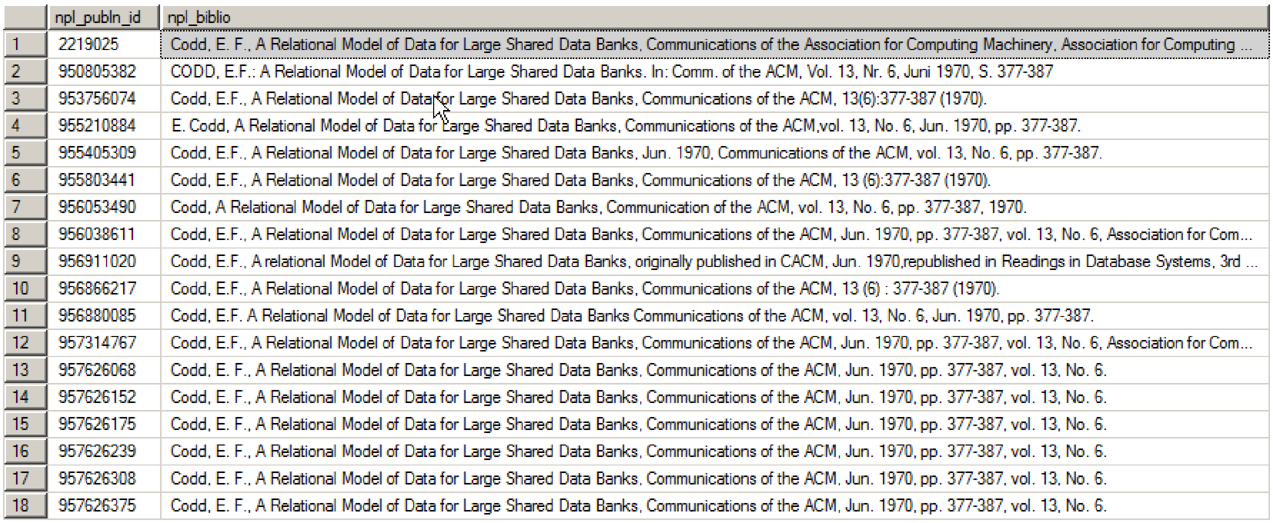
PATSTAT is a product of the European Patent Office (EPO)[4]. It is a periodical snapshot of patent related information organized in a relational database model. It contains records on patent applications, their applicants and publications. Table with a code name *tls214_npl_publn* (often referred to as TLS214) stores information on bibliographic references, like the one shown in Figure 1-A. The records, however, are often duplicated or inaccurate. Moreover, a full bibliographic reference is stored in only one attribute. This makes it problematic to query the table for relevant information, for example, to retrieve an author's name or the date of a specific publication.

1.1.3 Disambiguation of records

Disambiguation in the context of data management refers to the identification of unique entities within a dataset. Such entities are identified by a unique identifier that can be assigned to many database records. The database entries that effectively describe the same bibliographic entity are referred to as duplicates. The problem of data duplication and ambiguity arises due to (among other reasons):

a. Lack of consistent input (transcription) convention;
b. Variable level of input (detail) accuracy;
c. Missing data;
d. Different order of transcription of the same information;
e. Typos.

The Table 1-A illustrates the problem:



	npl_publn_id	npl_biblio
1	2219025	Codd, E. F., A Relational Model of Data for Large Shared Data Banks, Communications of the Association for Computing Machinery, Association for Computing ...
2	950805382	CODD, E.F.: A Relational Model of Data for Large Shared Data Banks. In: Comm. of the ACM, Vol. 13, Nr. 6, Juni 1970, S. 377-387
3	953756074	Codd, E.F., A Relational Model of Data for Large Shared Data Banks, Communications of the ACM, 13(6):377-387 (1970).
4	955210884	E. Codd, A Relational Model of Data for Large Shared Data Banks, Communications of the ACM,vol. 13, No. 6, Jun. 1970, pp. 377-387.
5	955405309	Codd, E.F., A Relational Model of Data for Large Shared Data Banks, Jun. 1970, Communications of the ACM, vol. 13, No. 6, pp. 377-387.
6	955803441	Codd, E.F., A Relational Model of Data for Large Shared Data Banks, Communications of the ACM, 13 (6):377-387 (1970).
7	956053490	Codd, A Relational Model of Data for Large Shared Data Banks, Communication of the ACM, vol. 13, No. 6, pp. 377-387, 1970.
8	956038611	Codd, E.F., A Relational Model of Data for Large Shared Data Banks, Communications of the ACM, Jun. 1970, pp. 377-387, vol. 13, No. 6, Association for Com...
9	956911020	Codd, E.F., A relational Model of Data for Large Shared Data Banks, originally published in CACM, Jun. 1970,republished in Readings in Database Systems, 3rd ...
10	956866217	Codd, E.F., A Relational Model of Data for Large Shared Data Banks, Communications of the ACM, 13 (6) : 377-387 (1970).
11	956880085	Codd, E.F. A Relational Model of Data for Large Shared Data Banks Communications of the ACM, vol. 13, No. 6, Jun. 1970, pp. 377-387.
12	957314767	Codd, E.F., A Relational Model of Data for Large Shared Data Banks, Communications of the ACM, Jun. 1970, pp. 377-387, vol. 13, No. 6, Association for Com...
13	957626068	Codd, E. F., A Relational Model of Data for Large Shared Data Banks, Communications of the ACM, Jun. 1970, pp. 377-387, vol. 13, No. 6.
14	957626152	Codd, E. F., A Relational Model of Data for Large Shared Data Banks, Communications of the ACM, Jun. 1970, pp. 377-387, vol. 13, No. 6.
15	957626175	Codd, E. F., A Relational Model of Data for Large Shared Data Banks, Communications of the ACM, Jun. 1970, pp. 377-387, vol. 13, No. 6.
16	957626239	Codd, E. F., A Relational Model of Data for Large Shared Data Banks, Communications of the ACM, Jun. 1970, pp. 377-387, vol. 13, No. 6.
17	957626308	Codd, E. F., A Relational Model of Data for Large Shared Data Banks, Communications of the ACM, Jun. 1970, pp. 377-387, vol. 13, No. 6.
18	957626375	Codd, E. F., A Relational Model of Data for Large Shared Data Banks, Communications of the ACM, Jun. 1970, pp. 377-387, vol. 13, No. 6.

Table 1-A Example of 18 out of 56 records found by a simple search on the exact title match. Thus, even more records referring to the same entity may exist in the database.

All of the records shown above refer to the same entity – the paper by E.F. Codd on the relational database model (Codd, 1970). However, the references to the same entity are given in different ways

[4] https://www.epo.org/index.html

or are simply duplicated. For example, record 7 does not contain Codd's name initials and the month information, while record 8 contains full transcription of the abbreviated name "ACM" at the end of the string. All the records describe the same bibliographic entity but are treated as distinct entities by the primary key of the TLS214 relation – the *npl_publn_id* attribute.

Such a design makes it very difficult to use information in the table in a correct way. For example, say a researcher is interested in the relation between science and technology. She assesses that a scientific discovery is well proxied by a publication of a scientific paper, while a piece of technology can be modeled as a patent publication. However, due to the unsupervised procedure in which citations are added to the PATSTAT database it is difficult for her to specify a query that takes into account all the possible variation in records that describe the same bibliographic entity. As a result, the researcher is unable to properly count all of the scientific references to the same bibliographic entity. This results in incorrect patent statistics. For example, when one tries to identify a single researcher and his body of work (like E.F. Codd). Also, studies on a population of researchers are difficult without a prior cleansing and de-duplication of the PATSTAT database.

1.2 Research Question

The goal of this paper is the disambiguation of scientific references in the *tls214_npl_publn* table of the PATSTAT database with use of an automated method. As a result, the research question is:

"*How to disambiguate scientific references in the PATSTAT database for the purpose of economic research and policy evaluation?*"

"Scientific references" refer to the types of records in the table that describe entities that can be classified as publications of theoretical or empirical origin. Not all records in the table are scientific references. While some cleansing and disambiguation measures can be applied to all records, the paper will specifically focus on providing the best results for scientific references. The final result of the procedure is a table with clusters of name variants (i.e. records) for each, unique scientific entity.

The term "PATSTAT database" is a synecdoche[5]. In particular we refer to the *tls214_npl_publn* table that stores information on non-patent literature references which can be found on patent application and publication documents.

"Economic research and policy evaluation" refers to the domain of applications for which the performed disambiguation procedure can be especially useful. Other uses, like data compression, are also connected to this project, however, they do not constitute the focus of the methods.

Throughout the paper other relevant sub questions arise:

[5] As in when I say „Miami won NBA Finals", I do not mean the whole city of Miami, but the basketball club "Miami Heat".

1. What types of documents are cited in the table?
2. What are the bibliographic conventions used?
3. What are appropriate methods for bibliography cleansing?
4. How to appropriately label information in the bibliography?
5. What measures to use to detect duplication of records?
6. How can the methods be evaluated?
7. What are the SWOT (Strengths, Weaknesses, Opportunities, Threats) of the employed procedure?

1.3 Research Relevance

We identify two primary reasons that make this project relevant: the need for effective measures for policy evaluation as well as by allowing for empirical research in economics of innovation.

Policy evaluation is an important procedure from the perspective of welfare economics. If we aim to assure that the well-being of the whole society is maximized we also need to have reliable measures that can inform us how efficient we are at this task (Stanford Encyclopedia of Philosophy, 2013). As a result, the allocation of public funds is not only a political problem, but also an economic one. Moreover, from the Public Finance perspective if public money is spent, such expenses must arrive at more efficient outcomes than what private provision of such good could achieve. This is because the collection of taxes is associated with transaction costs, and so, with efficiency losses [see: excess burden in (Rosen & Gayer, 2008, p. 331)]. Such spending can be justified if the efficiency gains achieved through government spending are significant [see: theory of the second best (Rosen & Gayer, 2008, p. 341)] (or if they allow efficient provision of public goods [see: public versus private provision (Rosen & Gayer, 2008, pp. 62-66)]. As a result, policy evaluation is crucial to inform decision makers if the supplied funds are justified. Despite the fact that this paper does not focus on evaluation of a specific policy, nor does it try to derive measures for such an evaluation, it nevertheless works towards providing a necessary environment in which such an evaluation is efficient and accurate.

The second importance of this work comes from understanding the relation between innovation, science, technology, industries and wealth. For example, innovation economics proposes that supply side of the economy cannot flexibly adjust to ever changing demand side without good incentives to innovate with new products (Mowery & Rosenberg, 1979). As a result, it is important to identify and study the obstacles in the way of innovation. This can be achieved, for example, through cohort studies of scientific publications and their development within a scientific community, related technology area, and finally, the associated industry. Despite the fact, that this paper does not attempt to propose a theory of how innovation is achieved (e.g. through adoption of technology based of scientific discoveries), it works towards establishing a dataset on which such theories can be empirically verified.

Additionally, the output of the employed procedure can be used by other database scientists, e.g. patent statisticians, to relate information from the de-duplicated table to other resources like the Web of Science database[6]. In this sense more advanced queries can be performed and, as a result, the set of testable theories on the available datasets can be expanded.

For a comprehensive study into the landscapes of projects that aim to evaluate the relation between science and technology please see (Winnink, Science-technology interaction (an annotated bibliography), 2015).

1.4 Research Methodology

The project uses the 2014 version of the PATSTAT *tls214_npl_publn* table to derive a sample of publications on which the disambiguation procedure will be performed. Due to computational limitations the complete dataset will not be disambiguated, however, it is possible to do so with enough computing power. Methodology is further developed in Chapter 3, and consists of the following stages:

1. Pre-Cleaning;
2. Cleaning;
3. Pattern Extraction;
4. Pattern Evaluation;
5. Pairing Rules and Scoring;
6. Obtaining Clusters.

The project was conducted on Microsoft SQL Server 2012. The codebase is written in T-SQL, that also utilizes CLR to perform string operations in C#. The technology and software used in this project is discussed in more detail in the Appendix.

1.5 Thesis Outline

The structure of the thesis is set as follows: Chapter 2 describes the data variation issues of the TLS214 table. Chapter 3 discusses in detail the methodology of the cleansing and disambiguation process. Chapter 4 presents the results, in particular, the performance of the technique as measured by precision and recall. It also discusses statistics on the discovered clusters. Chapter 5 concludes the paper and discusses limitations through SWOT analysis of the employed procedure.

[6] http://wokinfo.com/

2 DATA

This chapter presents the essential information about the PATSTAT database and the TLS214 Table in order to understand the context in which the disambiguation techniques are used.

2.1 PATSTAT database

PATSTAT is released (updated) in half-yearl intervals. The Figure 2-A below presents a part of the PATSTAT schema (EPO, 2015, p. 22), with the TLS214 table marked in a red rectangle:

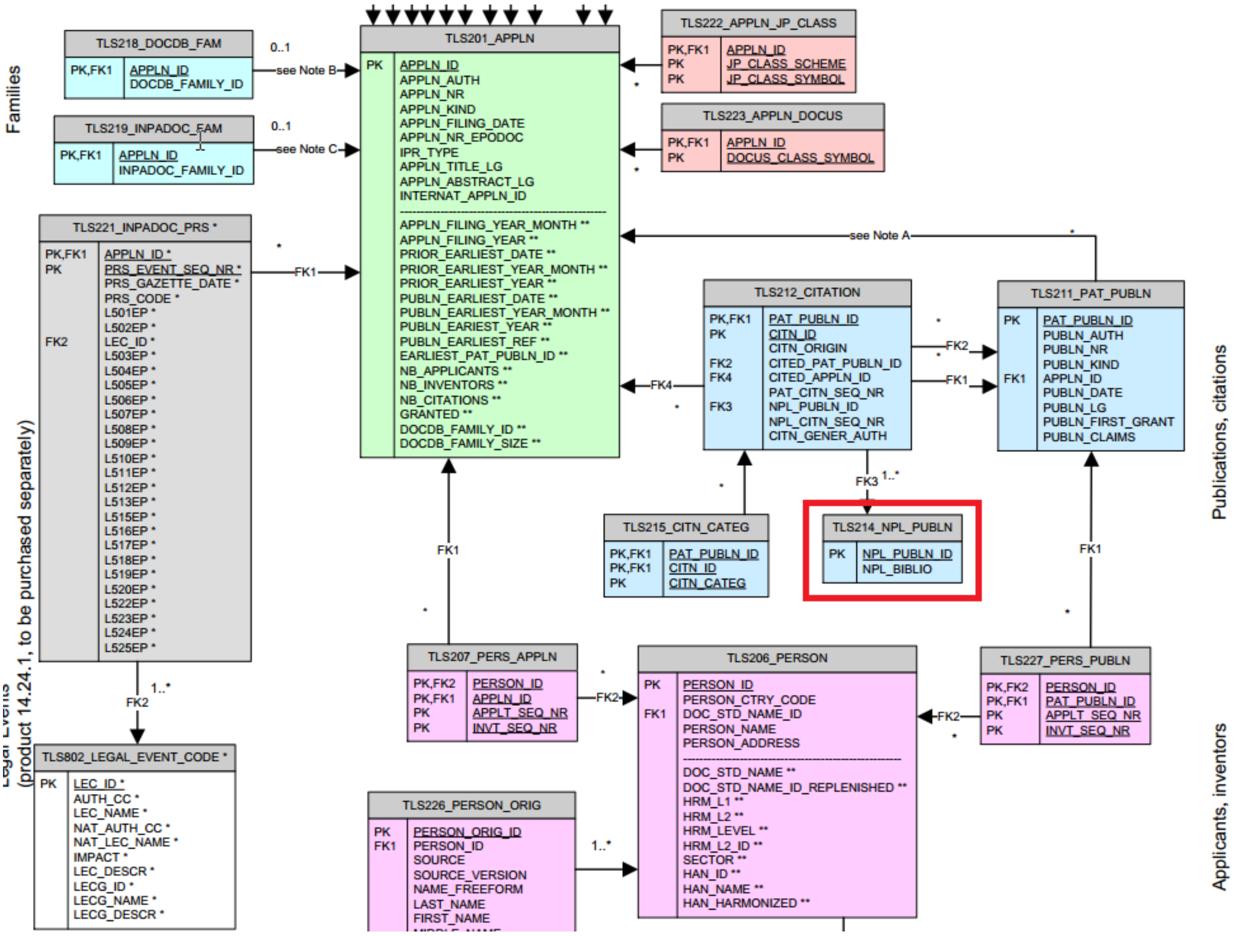


Figure 2-A Part of the PATSTAT DB with the tls214_npl_publn marked in red

There are three main “events” or “entities” described by the PATSTAT database:

1. Patent Application - TLS201;
2. Applicant - TLS206;
3. Patent Publication - TLS211.

The remaining tables either contain additional information about those entities (e.g. TLS202 for TLS201) or establish an entity-relationship (e.g. TLS207 between TLS201 and TLS206). The relational structure achieved in the PATSTAT database is derived from information stored in another

EPO system: DOCDB. DOCDB stores data in the XML format and, as such, can be characterized as "file processing system" (Silberschatz, Korth, & Sudarshan, 2006, p. 3).

2.2 Context of TLS214 in the PATSTAT database

Once patent application is considered by a patent office and the patent is granted a new entry to the TLS211 table is added. Patent publications (stored in TLS211) may contain references to other literature sources. There are two primary reasons to publish references to other resources in a patent publication. The first one is to show in what aspect applicant's invention is different from another invention with a published patent. The second reason is to indicate the sources of information used in the patent application, e.g. a scientific paper that develops a specific terminology, that is used in the invention description.

The structure of PATSTAT distinguishes between those two possible types of documents that can used as citations in a patent publication. Patent references are stored in the TLS212 Table, while Non Patent References are saved separately in TLS214. As PATSTAT keeps record of documents referenced by the TLS212 (i.e. patent publications or applications in TLS211 or TLS201) it is easy to verify the *referential integrity*[7] of this relation. The same, however, cannot be said about TLS214 table, because the entities that the bibliographic references are pointing to are not classified within PATSTAT. As a result, while PATSTAT is well suited to use information about citations to other patent applications or publications it does not provide a complete solution to efficient use of non-patent references.

2.3 TLS214 - Non Patent Literature - NPL Table

While in practice the NPL table does contain records that point to patent literature e.g. Search Reports (EPO, 2015, p. 56); it is, in principle, aimed to store references to entities that are not stored in the EPO system. As a result, the way bibliographic information is presented is in form of a single text string. This string is presented as it was provided by the applicant, without changes (Table 1-A).

EPO provides a way to disambiguate between NPL entities by use of "Non-Patent Literature reference number" (EPO, 2015), which will be referenced as the XP number[8]. The XP number allows to search for and identify unique bibliographic entities and investigate which patent applications cite them. The XP number can be (but need not be) specified by applicants in the bibliography field. If detected, it is used as a unique identifier of the TLS214 table in the *npl_publn_id* field.

[7] This means that all unique entities to which references point to can be found. As a result, all information about any cited patent publication (or application) can be retrieved.

[8] Due to the use of those code letters as a prefix to the numbers.

2.3.1 NPL_PUBLN_ID, NPL_BIBLIO

Table 2-A presents basic information about the TLS214 table. TLS214 is composed of two columns: a primary key attribute – *npl_publn_id* (i.e. a unique identifier of the relation) and *npl_biblio,* which stores string data about a bibliographic reference.

Fields:	**Number of records**	**Size (GB)**	**Source**	**Data Type**	**Domain**	**Range(s)**	**Default**
npl_publn_id	23,806,543	0.6	DOCDB, PATSTAT	Integer	0-999,999,999	0-950,000,000	0
						950,000,001-999,999,999	
npl_biblio	23,806,543	14	DOCDB	String	Max 3000 characters	-	n/a

Table 2-A Basic information about TLS214 Table of the PATSTAT database.

There are two ranges for the ID values:

1. Range: 0-950,000,000 – Reserved for documents with detected XP number. If XP number is detected in the bibliography, the *npl_publ_id* takes its value.
2. Range: 950,000,001-999,999,999. A *Surrogate key* created for all other bibliographies.

Various documents are referenced in the bibliography field. Among others they include:

1. Scientific papers published in scientific journals or magazines;
2. Books;
3. Standard specification documents;
4. Collections of abstracts of scientific papers or annotated bibliographies;
5. Legal documents (usually associated with certain application number);
6. References to search reports or other documents produced by EPO.

Because of the variety of described resources not all bibliographies contain the same information. However, PATSTAT documentation (EPO, 2015, p. 58) specifies that it is typical for records to include information about:

1. Author(s);
2. Title of the article;
3. An abstract;
4. Date;
5. ECLA Classification;
6. ISBN, ISSN or DOI number.

Those attributes are also typical for bibliographies that describe scientific papers. Since this paper focuses on the relation between technology and science, it must be remarked that the employed scientometric techniques will focus on disambiguation of scientific resources, rather than offer a complete solution to disambiguation of all resource types described in the TLS214.

2.4 Exploratory Data Analysis (EDA)

According to (NIST/SEMATECH, 2012) the goal of the EDA (among other things) is to:

1. *Maximize insight into a data set;*
2. *Uncover underlying structure;*
3. *Extract important variables;*
4. *Detect outliers and anomalies.*

As a result, the first step of the procedure is to familiarize with the dataset and the types of variation that distinguishes similar entities. We can identify two different types of properties that characterize the *npl_biblio* field of the TLS214 table. The first type of properties focus on string characteristics of the field. The second type of properties is looking into the content of the bibliography – i.e. we look for string patterns that can be observed in the dataset.

2.4.1 String characteristics

It is intuitive to think of EDA as discovering how records are different from each other. In the most basic way records differ by what sort of characters they use. Table 2-B shows some summary statistics on the string content of *npl_biblio*.

Panel B	**Text statistics:**	***Mean***
	Amount of (per record):	
npl_biblio:	Characters (string length)	130
	Digits	15
	Alphanumeric words	20
	Capital letters	18
	Special characters	29
	Punctuation {,.:;}	8

Table 2-B

Numeric characters describe dates, page ranges, volume & issue numbers, unique identifiers (like ISSN and ISBN) or other numeric constructs used to describe the resource. Numbers are quite common in bibliographies and, on average, constitute almost 12% of the string length. Moreover, numeric characters can be said to be much more specific than alphabetic characters. A single change to a digit (e.g. from “vol. 51” to “vol. 52”) across two records is more significant in conveying the fact that the records describe a different bibliographic entity, than a single change to a alphabetic character (which can be considered a typo, e.g. “vol. 51” and “bol. 51”).

Use of capitalization is significant, since in some cases a bibliography (or its parts) is written in all caps. However, a direct string comparison is case sensitive and would not detect such an obvious duplicate. From the perspective of information content of a record its capitalization pattern is not relevant to its description of a resource. However, capitalization patterns are used in to signal transcription of specific information like author names, a title or a proper name. As a result, in principle, capitalization patterns can be used to extract certain information from the bibliography.

Punctuation is most prevalently used to distinguish between tentatively delimited fields that describe different information. For example in a record: “John Smith, “Molecular Physics”, 1/1/1990” a comma is used to delimit fields that describe different information. Other punctuation marks like “;” can be used to delimit a field. However, more often than not, a delimiter mark is omitted while the punctuation mark is used in another way (e.g. a single name can be written as “Codd, E.F.”). This makes it problematic to conveniently split a bibliography into its constituent fields.

2.4.2 Content related

What is more significant about the *npl_biblio* field is what sort of information it contains. In principle, records can be written in three official EPO languages: English, German or French. In practice, however, most records are written in English.

Table 2-C lists frequencies of some recurrent patterns in string content (often referred to as “tags”).

Panel C	**Sample: 102440 records**		
	Content (mentions):	***Frequency***	***% of sample***
2) Strings [] – omitted in search; *Italics* for literals	*et al*	42775	**41.76**
	None	1684	1.64
	page	3011	2.94
	volume	343	0.33
	issue	506	0.49
	http	2665	2.60
	[See references] *of EP* [number]	1806	1.76
	&#X	8172	7.98
	DIN	159	0.16
	[*NICHT*] *ERMITTELT*	122	0.12
	German and French month names	1352	1.32
	"*journal*" or "*magazine*" or "*article*"	7046	**6.88**
	"*abstract*" or "*application*" or "*publication*"	7793	**7.61**
b) Abbreviations	English month names	16567	**16.17**
	pp [or] *p.* [or] *pgs*	27145	**26.50**
	vol [or] *vol.*	27545	**26.89**
	no [or] *no.*	27593	**26.94**
c) Identifiers	*XP* [number (NPL reference number)]	5777	5.64
	ISBN	594	0.58
	ISSN	2204	2.15
	DOI	562	0.55
d) Domain related[9]	*&* [Symbol – Delimits "Corresponding Documents" or author lists]	0	0.00

Table 2-C Literals used to test the dataset for containing certain information

The tags were chosen based on the fact that *npl_biblio* mostly describes resources like scientific papers that contain specific key words to indicate description of certain information. For example, the word "page" signals (although not determines) that the bibliography contains page information. However, there are many ways to indicate that the page range is being quoted. For example "pages" can be used when a page range is given, or the "pp." abbreviation is used when one wants to shorten the bibliography.

From further visual inspection of records 4 main sources of variation were evident to influence the problem of records disambiguation:

1. Order of information:
    a. Numbers: "vol. 3, no. 4, 1990" vs. "1990, vol. 3, no. 4";
    b. Words: [author] [title] [journal] vs. [title][author][journal].
2. Convention of transcription:
    a. Numbers: 12/3/1990 vs. 12.03.1990;
    b. Words: "Smith, J." vs. "SMITH J".

[9] (Winnink & Kracker, Multiple items in NPL_BIBLIO strings, 2015)

3. Typos:
    a. Numbers “vol. 13b” vs. “vol. 13v” since “v” character is right next to “b” character on a QWERTY keyboard;
    b. Words: “Jamie Smith” vs. “Jane Smith” since “mi” substring is visually close to “n” character.
4. Level of Detail (LoD):
    a. Numbers: 1990 vs. 12/03/1990;
    b. Words: “Molecular Physics” vs. “Molecular Physics: a Comprehensive Study of Contemporary Innovations”.

Moreover, it was observed that author names can be written in a multitude of ways depending on the assumed convention of transcription. Some of such conventions are formal e.g. the APA Style allows to easily identify author names. However, use of a single, consistent format is not enforced in the *npl_biblio* and, as a result, there is large variety of used conventions. The issue of format based extraction that explicitly addresses this problem is discussed in detail in Section 3.7 (p.30).

In conclusion, the usefulness of EDA resides in the way it suggests what sort of information is “*out there*” to extract. The following Methodology chapter discusses in detail how to efficiently extract information from the bibliographies by taking into account the different forms of variation that exist in the data. Such extracted and well formed data can be used to disambiguate records.

3 PROJECT METHODOLOGY

3.1 Software and technology

The tools and technology that is used to conduct the project is now briefly described. For more detailed explanation please see the Appendix 6.1. The project was conducted in SQL environment, with scripts written in T-SQL. Additionally, Regular Expression were used to find and extract patterns. In order to handle the Regular Expressions, C# programs were written and were used in T-SQL through CLR architecture of the .NET framework.

3.2 Method outline

We illustrate a step by step overview of the employed techniques that perform the disambiguation procedure. What is meant by the disambiguation procedure is that each record in a dataset (or a sample) is assigned to a so called “cluster”. Such a cluster is a set of records in which all tuples describe the same bibliographic entity. In practice, this means that we aim to assign to each ID of the *tls214_npl_publn* table (*npl_publn_id*) a corresponding cluster ID. The cardinality of this relation is that one *npl_publn_id* may belong to just one cluster and many different *npl_publn_id(s)* may belong to a single cluster[10].

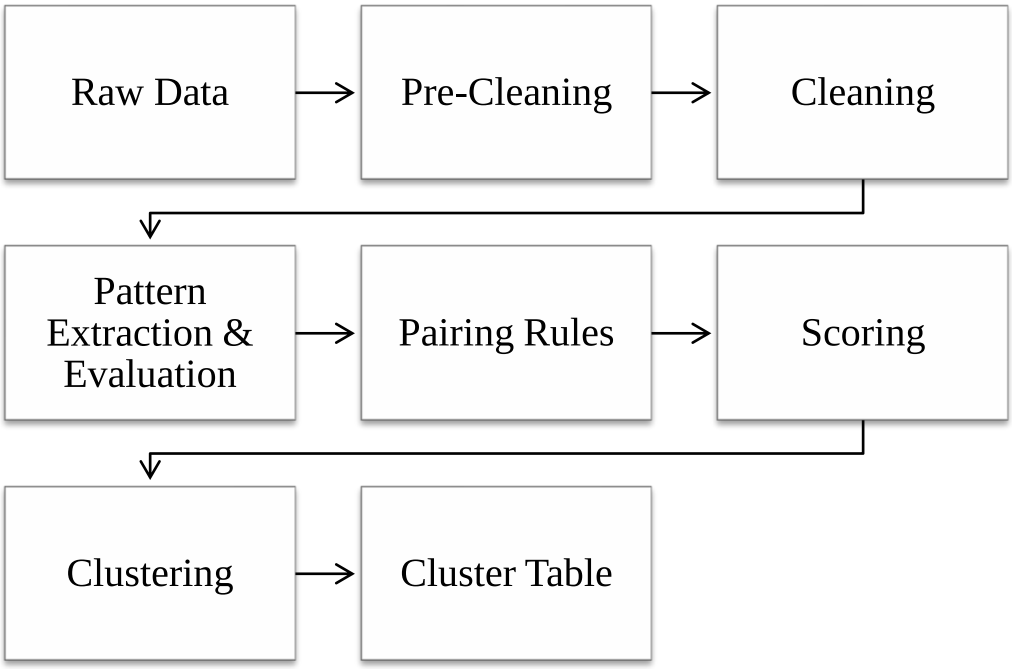


Figure 3-A A graph of major steps in the employed procedure

[10] In this sense, the presented disambiguation process is fully deterministic, what needs not to be the case for other disambiguation procedures. Use of fuzzy sets is an example of a probabilistic disambiguation technique.

The disambiguation process explicitly focuses on scientific records, which are only a subset of records in the TLS214 table. There is no easy way to identify records as the ones that describe scientific publications, and as a result, the employed procedures are performed on the whole sample.

It is believed that bibliographic entities that describe scientific publications allow for reasonable identification of the described entity through two "path ways"[11]. If one of the below listed methods fails, the other one can be used with high chance of success:

1. Identification by source:
   a) Journal (Publication source) name;
   b) Page(s) or page range;
   c) Volume;
   d) Issue;
   e) Publication identifiers (ISSN or ISBN);
   f) Date of publication.
2. Identification by author:
   a) Author(s) name(s);
   b) Author's origin or workplace;
   c) Description of the entity (e.g. title, abstract).

Most of the information on the above list is given in a complete bibliography of a scientific citation. As a result, the employed methods focus on extracting the above information from the records. Once such information is available, records can be compared with each other based on the way they exert similarity with respect to their extracted attributes. It is believed that above certain threshold value such comparisons produce pairs of records that are describing the same entity – i.e. they are duplicates. What proves to be the problematic feature of TLS214 is that it is difficult to extract those attributes from unorganized resource such as the bibliography field. However, a series of techniques (Sections 3.4-3.8) were introduced to facilitate efficient extraction of information that can be later used to detect duplicates.

3.3 Sampling

Panel A of Table 2 shows that there are almost 24 million records in the TLS214 table, what amounts to a total of 14.6GB of information. Computations on such a large amount of data presents significant computational requirements, that might translate to long processing times. As a result, it is more

[11] This is a very useful property of scientific references, what makes them a feasible subject for a disambiguation process that works with limited amount of unstructured information. However, other records in the TLS214, that for example use various unique identifiers, and so, have only one path way to be disambiguated, are not well suited to be disambiguated using the employed measures.

feasible to work with a representative sample of data, rather than the whole dataset. In principle, all the operations performed on such a sample can be applied to the whole dataset.

With the available hardware, a sample of 100.000 records[12] was chosen as large enough to reflect the variety of data, while keeping the computation time in a reasonable time frame. The employed sampling technique cannot be classified as a fully randomized procedure, but we believe that for the used purposes fully randomized design is not necessary.

3.4 Pre-Cleaning

In Section 2.4 we distinguished between string and content related features of the *npl_biblio* field. In line of this distinction, our cleaning procedures distinguish between those measures that modify string features of the bibliography field and those that interfere with its informational content. There are two reasons for such a division.

First is related to performance. Some string operations used for information extraction are considered heavy in performance. It is therefore sensible not to perform those operations on records that can easily be detected as duplicates (especially when their amount is substantial).

The second reason relates to the concept that interference with data content also modifies its informational content, and therefore such changes need to be transparent and prudent. In the Pre-Cleaning stage we perform operations we believe do not significantly change informational content of the records, but eliminate cases when slight differences in transcription (of virtually the same bibliography) are not registered as duplicates.

3.4.1 Whitespaces

A record: *"   Abcde"* has unnecessary leading whitespaces and is considered equivalent to the *"Abcde"* string. The same reasoning applies to records that contain whitespaces at the end of the string (referred to as lagging whitespaces). Also, all occurrences of double or more whitespaces within the string are considered informationally equivalent to a single whitespace. If a dot appears at the end of a string it is also removed.

3.4.2 Diacritics

While use of diacritics carries important informational value it is often the case that the accents are not properly transcribed (or omitted). This is especially the case when e.g. names are written by a foreign user. In order to account for this common change in transcription, letters that do contain special accents are replaced by their closest equivalents in the standard English alphabet. E.g. German character "ü" is be replaced by "u". As it is important to detect insignificant duplicates as early on as

[12] A T-SQL, built-in method was used to draw the sample. A user can supply his desired amount of records, but each time the sample is drawn a slightly different amount of records is retrieved (usually the difference from the sample to the specified value is no more than 3%).

possible, the diacritics are removed in the pre-cleaning stage. We believe that this measure should not significantly increase False Positive rate of our method, but can greatly improve the True Positive rate.

3.4.3 Capitalization

Some capitalization patterns in records include:

- The whole string is capitalized.
- A part of the string is capitalized to highlight important piece of information like author name or title.
- Every first letter of a title is capitalized.
- First letters of author names are capitalized.
- Initials or proper names use capital letters.

As a result, duplicate records can display significant variation in the way they are (not) capitalized. This results in a set of trivial duplicates, that differ only by the way words are capitalized. Pre Cleaning aims to detect those obvious duplicates, however, capitalization patterns are considered way more important bibliographic feature than to be simply removed from the dataset (like in the case of diacritics). As a result, at the final step of Pre-Cleaning the obvious duplicates were detected after applying the lowercase to all character. However, after detecting such duplicates, the original name forms (i.e. with capitalization) were reverted back to be used in further analysis. As a result, obvious duplicates were detected early on, and the requirement that the Pre-Cleaning cannot alter informational content of records in a significant way was also satisfied.

The main advantage of Pre-Cleaning is that obvious duplicates are detected early on, and those records are not processed further. However, not all duplicate records of the same bibliographic entities can be detected in this way. Cleaning stage discussed in the next section is the first stage in a series of steps that aim to detect duplicates based on the information stored in the bibliographic entries.

3.5 Cleaning

In contrast to Pre-Cleaning that focused on non intrusive techniques Cleaning stage is less strict in altering records' content. This is warranted by notion that the very same piece of information can be decoded in a slightly different way. Cleaning stage aims to identify those alternative ways in order to harmonize this variation to a common denominator. To this end, the use of regular expressions is invaluable. As explained in Appendix 6.1.3, regular expression parse a string in search for characters that follow a format (or literals) specified by the regular expression. Such a solution provides a flexible mechanism to detect various forms in which information can be transcribed. The process in which Regular Expressions are constructed can be summarized in the following steps:

1. Identify a commonly used label (a "tag") that signals transcription of specific information. For example, "page" label in the "Smith et. al, Nanotechnology, page 34" string signifies that number 34 demarcates page information.
2. Look for popularly used variants of the found label that are used in the same way as the standard form of the label. For example, common alternatives to the "page" label are "p.", "pp" or "pgs" tags.
    a. Investigate dataset specific label variants that may improve the overall harmonization rate of the dataset. For example, German word equivalent for the "page" label is "seite". Its plural version "seiten" is also commonly used.
3. Identify and account for commonly used transcription signs or formatting characters that accompany use of tags. For example, it is possible to write down abbreviation of "pages" label as "p" as well as "p." or "pp" and "pp.". The addition of the dot is optional. Also, one needs to account for spacing patterns before and after the tag to make sure tags are standalone words and not part of other words within bibliography[13].
4. Compose a regular expression that matches strings following the identified tag variants. To this end, assume case insensitive mode, since capitalization is not significant for labels harmonized in this step.

In the next subsections we provide a detailed discussion of harmonization steps aimed at some of the attributes listed in Section 3.2.

3.5.1 Source information

Numeric source information is relatively easy to detect since it is often accompanied by use of tags (e.g. "page 3, vol. 5, issue 4"). A similar case holds for dates that explicitly use month names to indicate date information.

For this reason, it is more difficult to find full journal name in the bibliography string, since it is not accompanied by a suitable tag. One solution to this issue is to compose a database of journal names and screen each bibliography to check if it contains a pre-defined journal name. Not only is such a solution too demanding on performance[14], there is also no guarantee of good results, since many journal names might not be listed in the database. A middle of the road solution is to look for certain

[13] While any direct occurrence of the "pages" substring in a record can be said to be satisfactory to conclude that the page information is indeed accompanied by such a label (we can say that "pages" is specific) it is more difficult to say so about the single character like "p". First, one needs to make sure that "p" tag is separated from other characters, as for example in "vol. 3, p. 6". Otherwise, label "p" would be matched within the string "picture" and changed to "pagesicture" if the used harmonized form is "pages". Other step is to only allow conversions of matches that are actually followed (or proceeded) by numbers. This measure avoids a case when "p" tag in "John P. Smith" is converted to "John pages Smith".

[14] One tested database contained 1352 journal names. Direct string comparisons between bibliography field and this database were very time consuming, even on a small samples of 10'000 records. The outcome of this extraction method did not produce significant extraction rate to warrant use of such a method.

key words that often appear in journal names or elsewhere in the bibliography. Those keywords can be used as proxy of what sort of information is described.

Table 3-A presents only a sample of examples of the types of label variants that were harmonized in the Cleaning Stage. Transcription signs and formatting characters were not listed in the Allowed Forms column to avoid repetition of information, but they were accounted form during the cleaning procedure. A full table with all used harmonized forms and their allowed formats is listed in Appendix Table 6-A:

Type:	**Harmonized form**	**Allowed forms**	**Context sensitive**	**Symmetric**
Source	*pages*	*p*	Y	N
		pp	Y	Y
		pgs	Y	Y
Bibliographic Type information	*abstract*	*abstr*	N	N
		abstract	N	N
	magazine	*mag*	N	N
		magazine	N	N
	jour	*jour*	N	N
		journal	N	N
Date: Month names	*jan*	*jan*	N	N
		january	N	N
		januer	N	N
		januier	N	N

Table 3-A Excerpt from Appendix Table 6-A, that presents what sort of substring variation was harmonized to a common form.

- “Type” column describes categories of harmonized labels.
- “Harmonized form” column shows to what format all positions in the ‘Allowed Forms’ list are changed to.
- “Context sensitive” attribute indicates whether the used regular expression checks for other information around the Allowed Form. E.g. for the “pages” label the RegExp only alters those occurrence of “p” label that are followed by a number.
- “Symmetric” column indicates if the context condition is checked on both sides of the label – i.e. “p” is not symmetric therefore 6 in “6 p” would not be classified as page 6. However “pgs” is symmetric, therefore 6 in “6 pgs” would be matched as page number.

3.5.2 Author information

Author names are more difficult to identify than source information since the names are not accompanied by a label like *author*. The closest equivalent to such a label is use of *et. al* tag that signifies that some author names were omitted from the bibliography. *et. al* is positioned after specifying the primary author of a publication, therefore making it possible to extract alphabetic characters to the left of the *et. al* tag as information about primary author’s name.

Moreover, many bibliographies simply begin with listing the author names that are separated from fields with other information by a delimiter like a comma or semicolon. As a result, we decided to harmonize all semicolons in a dataset to a comma, since the change in meaning is minimal. This step allows to obtain Type E format based name as outlined in Table 3-C.

3.5.3 Negative labels

Some so called "Negative labels" were also harmonized to facilitate further identification of records that are not scientific records. For example, records that contain an application number (harmonized to *appln no* string) cannot be properly disambiguated other than by using that specific application number. As a result, the employed disambiguation method (that focuses on comparing features of scientific bibliographies) should be applied carefully to those records. Negative labels limit the extent to which our method is applied to records that are not well suited for such an approach.

3.6 Pattern Extraction

The Cleaning stage has prepared the dataset to parse the records in search of specific tags that describe bibliographic information[15]. Two types of regular expressions were used to extract information.

The first type is characterized by use of some sort of literal characters (like a tag) to narrow down the search. An example is the *month_date* attribute, that looks for month names followed by numbers that either describe day of the month or year information. Such regular expressions are unlikely to produce erroneous results, i.e. to extract information that has nothing to do with what was intended to be extracted.

The second category of regular expressions describes a specific *format* in which information is conceived to be written, while it is quite possible that substrings of characters that will be matched by the RegExps are unrelated to their intended attribute. The primary example of such regular expressions are Format Based Names (FBNs) that are discussed in detail in Section 3.7.

Table 3-B (which is presented in full in the Appendix Table 6-B) describes examples of attributes that were extracted:

- "RegExp Group" column describes what sort of information is extracted.
- „Label" column is an internally defined name of the attribute.
- "Subject" column specifies what sort of phenomenon is the RegExp primarily focusing on.
- Constraint column states a more precise definition of what sort of condition is enforced on the subject of the Regular Expression.
- "In-between characters" column lists transcriptions signs and formatting characters that are optional and may be embodied in valid match.

[15] Some other patterns that did not require any harmonization steps and are also extracted in this stage.

- “Outcome” column declares what sort of output is produced by a match of a regular expression.

Table 3-B Excerpt from Appendix Table 6-B showing examples of Extraction Patterns

RegExp Group:	Label	Subject	Constraint	In-between characters	Outcome	Example 1	Example 2
Date	*month_date*	{Month name} labels	Followed by a number	Whitespace (optional)	Month name and number	may 14	may 2009
	tentative_easy_year	Numbers	Between 1850 and 2015	None	Four digit number	1993	2014
Source	*easy_pages*	"pages" label	Followed or proceeded by a number or number range	"-" or " to " or "/" or Whitespace	Number or number range	6	56-59
	easy_volume	"vol" label	Followed or proceeded by a number	Word boundary	Digit(s) or digit(s) with letters	3	3a
Other Properties	*s_start*	Start of the string	8 characters	-	8 characters	"Smith J,"	“Physics “
	s_end	End of the strong	8 characters	-	8 characters	"a, May 1"	
	bib_numeric	Numbers	Preserve single space between numbers	Any	All digits in the original string	22 221 200 292 109 200	26 132 003 297 303
String properties (RegExp not used)	*sum_of_num*	Calculates sum of all numbers in the string	-	-	Integer	12569	
	count_of_num	Calculates count of all numbers in the string	-	-	Integer	8	

The Pattern Extraction stage focuses on capturing all reasonably popular patterns in which information is transcribed, with limited consideration about actual accuracy of such an extraction. This does not mean that the expressions used to extract those attributes need not be precise. This means that at this stage we yet do not assign to attributes any measure of their reliability. For example, the *tentative_easy_year* attribute is less reliable than *month_date* property. This is because the latter will only extract a number that is following a month name, rather than any number (as in case of *tentative_easy_year*) that is in reasonable range of possible year dates. Evaluation of attributes (along with their further cleansing and harmonization) is performed in Section 3.8.

3.7 Hierarchical, Format Based Author Name Extraction Model

Although use of the "et. al" tag can be considered a successful attempt to capture author names in this specific dataset, it is problematic to consider this a complete solution to the problem of author name extraction. Some recent bibliometric literature is devoted to this problem and focuses on format based author name extraction (see bibliography 1-7 from (Constans, 2009)).

This section focuses on format based author name extraction and develops theoretical background for the system's subsequent implementation. In essence, such background is not necessary to the information extraction procedure, that can successfully derive its usefulness and validity from empirical performance. However, we believe that fundamental understanding of the modeled phenomenon is useful for understanding what types of retrieval methods to use (especially when one performs research).

The idea of format based extraction is based on an observation that sometimes the very specific way in which names are transcribed generates (within larger body of a bibliography) a phrase that is unique within that string based on its format. In principle it is then possible to parse the bibliography for that format, and since it is a unique format, only the name information would be matched.

In practice, format based extraction is error prone, since there is no guarantee that the format in which a name is transcribed is indeed unique within a bibliography. Moreover there can be multiple formats, across different bibliographies, that all differ in the way they transcribe the same name. However, this paper makes an attempt to devise a Format Based Extraction Model and use its subsequent implementation to extract more author information.

We consider this model as a "conceptual exploration" and believe it can only be useful in case it can be shown that the *target system* (*npl_biblio* from TLS214) *is the system that the model defines*[16]. We do not, however, make such empiric tests and justify the use of this model on an assessment that the formats we extract are well grounded by the stylized facts of the model.

[16] I borrow this view of why models are useful from work of Hausman on philosophy of economics.

The model is presented in full in the Appendix 6.2. It is based on template based model of (Constans, 2009).

Based on properties described in stylized facts of the model we believe that the schemas listed below are the most preferred for author name transcription:

1. Name, Initial, Name – The *2nd non basic schema in natural order* – Type A[17].
2. Initial, Name – The *3rd basic schema in a natural order* – Type B.
3. Name, Initial – The *3rd basic schema in the reverse order* – Type C. Is identical to Type B with only change in order of first and second name (which is equivalent to change in order of $[nN]$ – word phrase and I - an initial character).
4. Name, Name – The *3rd trivial schema in natural order* – Type D. A restriction is made to accept names with arity (length) of no longer than 3.

We do not conduct any empiric tests for the prevalence of those formats. As a result, we simply look for those four formats across the bibliographies based on face validity of those schemas with respect to how well they reflect the concepts described in the stylized facts.

There is no guarantee that what we extract are in fact names, nor that they do in fact represent the instances of schemas we have described above. The heuristic is that the users that fill in the dataset (patent applicants) do follow (to a lesser or greater extent) the stylized facts we have outlined. This results in a significant sample of phrases that can be considered names purely based on their format.

The extraction process uses auxiliary regular expression to improve accuracy of the extracted matches. Table 3-C explains in detail the formation of each format. Auxiliary expressions detect only those formats (of their primary schema) that follow minimal arity that is possible for that given schema. For example, Type A1 auxiliary expression matches only those substrings that have a single $[nN]$ word at the front, a single I in the middle and a single $[nN]$ at the end of the expression. Type A regular expression is composed so that it captures up to 3 $[nN]$ and up to 3 I, in wherever count they occur[18]. Since Type A is a superset of A1, if A1 match is not detected, but the Type A regular expression nevertheless produces a match it must be considered an error.

[17] It is in fact a further assumption on the *2nd non basic schema in natural order* that I is in-between the two $[nN]$ phrases – one describes the first name, the other the second name.

[18] One can say regular expressions, like Type A, are *greedy*.

Table 3-C Format based names. All Format Based names listed in the Appendix Table 6-C

RegExp:	Label	Format	Constraint	In-between characters	Outcome	Example 1	Example 2
Format based:	*nameA*	$[nN]\{1,3\}I\{1,3\}[nN]\{1,3\}$	Only valid if the corresponding auxiliary match of nameA1 is not null	Whitespace, "-", "."	3 part name with variable count of terms per each part	John Adam S SMITH	John S.A. Wright-Philips
	nameA1	$[nN]I[nN]$	Case insensitive, but lowercase names must begin with a capital letter	Whitespace, "-", "."	3 part name with single term per part	John S. Smith	Patric J ADAMS
	nameB	$I\{1,3\}[nN]\{1,3\}$	Only valid if the corresponding auxiliary match of nameB1 is not null	Whitespace, "-", "."	2 part name with variable count of terms per each part	J.A.P. Smith	J. Adam SMITH
	nameB1	$I[nN]$	Case insensitive, but lowercase names must begin with a capital letter	Whitespace, "-", "."	2 part name with single term per part	J. Smith	J SMITH
	nameC	$[nN]\{1,3\}I\{1,3\}$	Only valid if the corresponding auxiliary match of nameC1 is not null	Whitespace, "-", "."	2 part name with variable count of terms per each part	Smith J.A.P.	Adam SMITH J.
	nameC1	$[nN]I$	Case insensitive, but lowercase names must begin with a capital letter	Whitespace, "-", "."	2 part name with single term per part	Smith J.	SMITH J
	nameD	$[nN]\{2,3\}$	Words have to have at least 3 characters	Whitespace	2 to 3 part name with no initials	JOHN SMITH	John Adam Smith
String property based:	*nameE*	Alphabetic words at the start of the string	Before first comma or colon	Dot	String of words	“John Patric; Pavel Colins, (...)”	

$\{x,y\}$ Quantifier is specifying how many time is a particular element allowed to repeat itself.

3.8 Pattern Evaluation

3.8.1 Attribute cleaning and harmonization

Pattern evaluation cleanses, harmonizes and unifies information extracted in Pattern Extraction stage (the one exception is author name field for which two attributes are preserved). Some steps are analogous to the procedures performed in the Pre-Cleaning stage:

1. Removing leading and lagging whitespaces;
2. Removing multiple whitespaces;
3. Removing special characters from alphabetic attributes;
4. Applying lowercase to alphabetic characters.

Some new basic steps are:

1. Convert data-types of attributes that store numeric characters to a numeric data-type (e.g. integer).
2. Devise a *residual* attribute that contains all characters not extracted by other attributes.
3. Set to *null* all fields that are empty.

Attributes that attempted to extract the same information (e.g. *tentative_easy_year* and *month_date* both could extract the same information) are now unified to a single field based on reliability of those attributes.

3.8.2 Date

The first step of the date attributes evaluation was to convert month names contained in the *month_date* label into numeric equivalents (e.g. “jan” is converted into 1). The next step is a unification of all extracted dates into one attribute estimate. The most reliable patterns for date extraction are the ones based on strict, systematic formats: the American, European or Japanese date format. If one of those formats was detected it took priority above other date estimates. The second most reliable date estimator is *month_date* label, that either extracts month and day information or month and year information. Finally, if year is not detected in any other way, *tentative_easy_year* attribute is used as a final year estimate.

3.8.3 Format based Names

Format based name labels are unified to a single field that provides the best estimate of a format based name for a given bibliography field. Type A name is used first, since this format tends to extract name with highest arity. Based on observed performance, the second priority is given to Type E name format, that captures names that are assumed to be listed at the start of the string. Next, since Types B and C are equivalent, the format with higher arity is chosen as the preferred match[19]. Finally, if no other formats are found, Type D name is used in the unified *name* attribute. This is because Type D is the least unique of the modeled formats and majority of the matches it extracts are False Positives.

3.8.4 Source

The label *easy_pages* is further broken down into *pages_start* and optional *pages_end* if a page range is detected. XP number is harmonized to a format that begins with a literal 'XP' that is immediately followed by string of 9 digits with leading zeros if necessary. Also only the number value of the XP number is extracted and stored in a numeric data type attribute.

The result of Pattern Evaluation stage is a table that contains the most reliable estimates of extracted attributes and attributes that describe a different bibliographic property of a modeled entity.

The example below presents some of the evaluated patterns obtained from a bibliographic string of a record:

d_day	*d_month*	*d_year*	*pages_start*	*pages_end*	*volume*	*issue*
2	1	2009	254	264	284	1
xp_number	*issn*	*bibliographic_type*	*sum_of_numbers*	*count_of_numbers*	*aetal*	*name*
2511116	ISSN: 0021-9258	jour	154	56	agopian audrey	agopian audrey et al

Example 3-A Evaluated attributes of a record

new_id	*npl_biblio*
2182	AGOPIAN AUDREY ET AL: A New Generation of Peptide-based Inhibitors Targeting HIV-1 Reverse Transcriptase Conformational Flexibility, JOURNAL OF BIOLOGICAL CHEMISTRY, AMERICAN SOCIETY FOR BIOCHEMISTRY AND MOLECULAR BIOLOGY, US, vol. 284, no. 1, 2 January 2009 (2009-01-02), pages 254-264, XP002511116, ISSN: 0021-9258, DOI: 10.1074/JBC.M802199200

Example 3-B Bibliographic string of a record

3.9 Pairing Rules and Scoring

3.9.1 Explanation of Pairs and Rules

A "*pair*" is a set of two records. Those records are in some aspect(s) alike. Pairs are found by comparing records to each other according to some similarity measure defined by a *rule*. Not all pairs match together duplicates of the same entity, but many pairs produced by strong rules produce pairs that are indeed duplicates. An example of a T-SQL code that obtains pairs from a rule is provided in the Appendix Section 6.3.1.

The most basic similarity measure is a Boolean value comparison of one of the evaluated labels. For example, all pairs of records that match on their *year* attribute are similar to each other in this one respect and may be considered as a useful pair. Obviously, pairs generated by a rule like the one in the

[19] Since formal implementation of arity condition is time consuming a condition based on string length of the name can be considered equivalent.

example are not satisfactory to detect duplicates of the same bibliographic entity. As a result, the atomic rules (the most basic rule constructs) are combined together to form composite rules.

3.9.2 Two Types of Atomic Rules

Section 2.4.2 illustrated that variation in bibliographic information can be distinguished between numeric and alphabetic characters. Table 3-D classifies atomic rules into two classes:

1. Alphabetic (word) rules (W) try to capture variation in text formations.
2. Numeric rules (N) aim to capture variation in numeric data.

However, some individual attributes need to be combined at this stage to form useful atomic rules:

Alphabetic atomic rules	**W**	**Used attribute(s):**	**W**	**Superset counterpart:**
	1a	*bib_alphabetic*	1b	Partial LD(*bib_alphabetic*)[20] and *s_start* and *s_end*
	2a	*aetal*	2b	LD(*aetal*)
	3a	*name* and *aetal* is *null*	3b	LD(*name*) and *aetal* is *null*
	4	*bibliographic_type*	-	-
	5a	*residual*	5b	Partial LD(*residual*)

Numeric atmoic rules	**N**	**Used attribute(s):**	**N**	**Superset counterpart:**
	-	*XP number*	-	-
	1	*bib_numeric*	-	-
	2	*ISSN* or *ISBN*	-	-
	3a	*pages_start* and *pages_end* and *volume* and *issue* and *d_year* and *d_month*	3b	*pages_start* and *volume* and *d_year*
	4	*sum_of_numbers* and *count_of_numbers* and *count_of_numbers* > 6	-	-

Table 3-D Classification of attributes into two classes of atomic rules

Some of the atomic rules (with a subscript "a") have a corresponding "Superset counterpart" (with a subscript "b"). Superset is a set that contains another set. This means that all pairs generated by applying an "a" rule are provided in the result set of applying the "b" rule. B rules are used to expand the "coverage" of our method – the amount of records that can be compared and paired with other bibliographies.

[20] Levenshtein distance was not calculated on the whole length of *bib_alphabetic* and the *residual* attributes, but on a 10 length sample substring "cut out" of the middle of those attributes. Levenshtein distance calculated on *name* and *aetal* attributes utilizes the whole attribute.

For example, consider the *aetal* attribute (that collects names extracted based on the *et. al* tag). Assume an author has a middle name that is sometimes not transcribed as an initial when her publication is published. As a result, there is a slight (one letter) difference between one *aetal* attribute extracted from one bibliography and the other *aetal* attribute found in another record. Rule 2a would compare those two attributes directly and since they are different a pair would not be created. In contrast, Rule 2b uses *Levenshtein distance* (transcribed as LD) to compare attributes and produce pairs. Intuitive definition of Levenshtein distance is the amount of edits one needs to perform to change one string into another string. In our example this value is 1, since the only needed edit is to delete the one added initial. Our implementation of Levenshtein distance uses the edit distance to calculate (in %) how similar is one string to another string. For example, "JA Smith" (length: 8) and "J Smith" are different by 1 edit, therefore "J Smith" is in (1-1/8)=87,5% the same as "JA Smith". The use of LD is based on the premise that pairs that score above certain threshold value can be considered as similar, and are therefore paired. We select this threshold based on observation of a cut-off point after which low LD values do not provide any evidence for the two attributes to be similar.

Use of Levenshtein distance is valuable since it also allows to compare similarity of records for which no attributes were extracted. This helps to detect duplicates in case our method fails to extract some bibliographic information or, what is more often, in case a record does not contain the attributes we aim to extract. Since our method focuses on scientific references we expect to extract the attributes and as a result aim to use Levenshtein distance sparingly, as it is compute heavy.

3.9.3 Obtaining Rules and Scoring Pairs

The first principle of composing composite rules is that best results are achieved when numeric and alphabetic variation is controlled for in the same composite rule. Atomic rules specify certain similarity characteristic between two records. A composite rule combines two or more of such characteristics to propose a *pathway* through which a duplicate may be detected. This means that rules are composed of restrictions on numeric and alphabetic properties of a record. While composing rules we also use a second principle, which states that:

A composite rule has to have at least one Strong or one Middle rule

Classification of rules into Strong, Middle and Weak categories is given by Table 3-E:

Rule Classes[21]:	**N**	**Score**	**W**	**Score**
Strong:	1	9	1a	9
	2	7	1b	8

[21] XP number was not classified since it is a unique entity identifier

			2a	7
			3a	6
Middle	3a	6	2b	5
			3b	4
Weak	3b	3	4	3
	4	1	5a	2
			5b	1

Table 3-E Three different classes of rules and their corresponding scores

The application of the rule creation principle results in the following set of double rules:

$$\{N\colon 1, 2, 3a\} \times \ \{W\colon 1a, 1b, 2a, 3a, 2b, 3b, 4, 5a, 5b\} + \{N\colon 3b, 4\} \times \{W\colon 1a, 1b, 2a, 3a, 2b, 3b\}$$

Where × is a Cartesian product between the two sets of rules, as enclosed by {} brackets.

Each rule in Table 3-E has an associated Score value. Those numbers are used to grade pairs generated by a rule. Scores of pairs generated by composite rules are obtained by adding the scores of the atomic rules they contain. For example, pairs obtained from the $N1W1a$ composite rule score 9 points from the $N1$ rule and 9 points from the $W1a$ rule. The total score of pairs obtained from this double rule is 18.

A unique pair may be scored by multiple rules. For example a pair may score on rule N2W3a and N3aW4 to obtain a total of (7+6)+(6+3)=22 points. However, one needs to make sure not to "double-score" pairs that are obtained from applying an "a" and "b" rule of the same type. For example, pairs generated by the $N1W1b$ rule contain a subset of pairs that is a result of applying the $N1W1a$ rule. This subset must be excluded from being scored by the $N1W1b$ rule since it was already assigned a score by the $N1W1a$ rule.

Moreover, one "Negative" rule was constructed. Overall, there were two, so called, "useless" Boolean attributes obtained in the attribute extraction stage. Those two attributes attempted to identify those records which were unlikely to be scientific references. Those records, or rather the pairs they may form, are subsequently punished in the scoring stage by subtracting from their sum of scores a value that is given by the following formula:

$$Negative\ points = -(Negative\ pairs\ pass\ point - Threshold)$$

"Negative pairs pass point" variable is a selected sum of scores that a non scientific record should achieve in order to be sure it can be safely used in the clustering procedure. As a result, all pairs that contain a non scientific record, but score equal of above the "Negative pairs pass point" will be disambiguated by the clustering procedure.

Once all pairs are obtained, a sum of scores that a unique pair achieved is calculated. The higher the score of a pair the more evidence there is that the pair has matched two duplicates. Based on observations of pairs it was decided that the score threshold above which pairs can be considered duplicates is 14. It must be acknowledged that we do not use any formula to devise this threshold, nor do we use a standardized technique to assign score values to atomic rules. Those values are selected based on evaluation of *relative* strength of rules to identify duplicates and the overall performance of the double and triple rules to detect duplicates.

3.9.4 Pair Namespaces

The modular design of the pairing rules allows to conveniently separate different classes of rules from being executed in the disambiguation process. As a result, we define Namespace "a" rules that contain double rules for which the W class uses the "a" subscript. Namespace "a" also contains the double rules that use W4 atomic rule. Namespace "b" rules contains double rules for which the W class uses the "b" version of the atomic rule. Such a division is useful, because it allows to focus on obtaining pairs from just one type of rules. This design especially comes in handy as the computation time to obtain pairs is very large and one may wish to skip on some rules if the utmost precision of the method is not mandatory.

3.10 Obtaining clusters

3.10.1 On Clusters

Cluster is a meaningful or useful group of data points (Tan, Steinbach, & Kumar, 2006, p. 487). In the context of disambiguation of scientific references a cluster represents a set of records that all describe the same, unique bibliographic entity. Since members of a cluster describe the same entity they must exert a unique kind of similarity that is, in principle, not displayed by any other cluster.

There are multiple clustering techniques that work with different types of data. A technique that we use is called Connected Components. Intuitive definition of connected components is that if A is paired with B and B is paired with C, while no other record connects to A,B or C, then the A,B,C sub-graph forms a cluster.

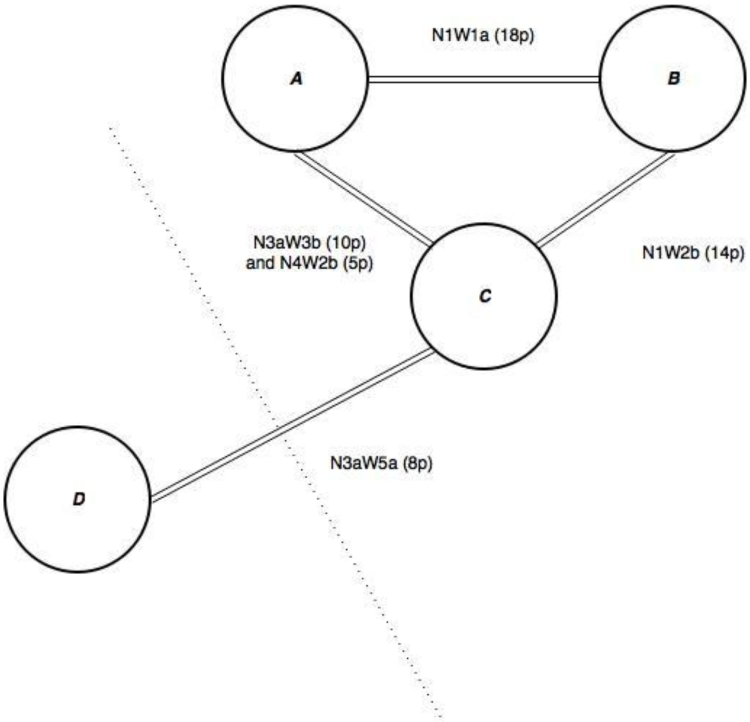


Figure 3-B Illustration of Connected Components subgraphs. Only ABC cluster is above 14 point threshold and the D record is not added to the cluster.

Another way of thinking about clusters is that they are an analyzed form of the obtained pairs (that scored above the threshold value in the rule construction stage). Pairs that contain common components are connected with each other to form a larger cluster. This network represents a unique bibliographic entity obtained across records that are duplicates of one another.

3.10.2 Coverage vs. Certainty

Clustering analysis needs to account for two issues in the process of classifying data points into clusters. The first is the problem of "coverage" – what amount of records can we test for being a duplicate of another records? In general, the greater the coverage the smaller the precision of extracted information. As a result, in our method we try to extend the coverage by employing the superset counterpart ("b") rules, while maintaining the atomic rules at quite a strict level.

Other limitation is related to performance. Certain comparisons were omitted from analysis since they can take too much time to compute[22]. This measure lowers the precision of our method, but we believe the overall accuracy was not affected.

In general the more diverse rules one can use, the better the coverage and the better the differentiation between different pairs with respect to their total score. However, one needs to be careful not to use

[22] For detailed overview of the omitted rules, please look into the codebase of the project, available at: https://github.com/blackwhitehere/TLS214-Disambiguation or verify omitted statistics in the Appendix Tables 6-D.

too many rules that in fact can be easily satisfied by virtually any pair. In such a case, pairs that are not duplicates obtain overestimated scores that may push them over the threshold and increase the overall false positive rate of the method. As a result, we believe that each composite rule has to hold some form of premise that the pair it generates is an actual duplicate. This is reflected in the setup of our method by limiting the composition of rules to only those that hold at least one Strong or Middle rule.

3.10.3 Post processing

Clustering algorithm assigns to each *npl_biblio_id* (unique identifier of the TLS214 table) a surrogate key ID of a cluster to which it belongs. Clusters are not described in any other way than through the set of records that belong to them. In order to finalize the clustering on the whole sample two steps need to be performed.

First, records for which no duplicates were detected (and as a result to which no cluster ID was assigned by the clustering algorithm) are assigned to new, single record clusters. This assures that if in the future a new, duplicated record is added to the database, then no new cluster needs to be created and the duplicated record can be appended to the already existing cluster.

Second steps addresses the obvious duplicates detected in the Pre-Cleaning. After Pre-Cleaning a representative of a group of duplicates was passed on to the next stages to see if other, non-obvious duplicates can be detected. Once the clustering algorithm has finally assigned the representative record to one of the clusters, the remaining duplicates from the Pre-Cleaning stage can be appended to the same cluster.

4 RESULTS

4.1 Processing times

Any computation intensive project, like the one presented in this paper, needs to be evaluated based on the performance of the employed methods. The Table 4-A presents processing times for each of the stages of the project[23]. The code was executed on a PC with 64 Bit version of Windows 7, 4GB of RAM and Intel i5 760 2.8G Hz processor.

Sample:	**102'440 or 99074**	**Time (h:m:s)**
1	Pre-Cleaning	00:00:56
2	Cleaning	00:01:23
3.1	Easy Labels	00:01:15
3.2	Format Labels	00:01:49
3.3	Evaluated Labels	00:00:33
4.1	Pairing Rules - Namespace a	02:18:13
4.2	Pairing Rules - Namespace b	01:23:18
5.1	Clustering	00:00:10
5.2	Post-Processing	00:00:17
	TOTAL	**03:47:56**

Table 4-A Processing times for each stage of the procedure

It is evident that the computation of pairs based on conditions specified by rules is the most computationally demanding process of the whole procedure. The explanation behind this can be seen in that if a record contains an attribute, the value of that attribute needs to be compared with other values of that same attribute, that belong to all other records. As a result, the greater the efficiency of the pattern extraction procedure, the more comparisons can be made between bibliographies and therefore the longer the computation times.

4.2 Cleaning and extraction statistics

For each (extracted and evaluated) attribute we present the amount of records (and corresponding percentage) from which a value was found. Also, we specify the overall amount of records affected by the methods applied in the (Pre) Cleaning stage:

[23] Times are likely overestimated since the result set was often times asked to be displayed, what is not a necessary step to perform the computation.

Sample: 102,440		**Duplicates detected**	**%**
Pre-Cleaning		3366	3.29
Sample: 99,074		**Records affected**	**%**
Delimiter	*;*	11891	12.00
Source	*pages*	41313	**41.70**
	p [number]	1423	**1.44**
	vol	2850	2.88
	no	26011	26.25
Author	*et. al*	40605	**40.98**
Bibliographic labels	*proc*	3704	3.74
	science	1839	1.86
	chem	6095	6.15
	natl	1517	1.53
	appln	10910	**11.01**
	publn	554	0.56
	artl	442	0.45
	abstract	2288	2.31
	magazine	130	0.13
	jour	5648	5.70
	pct	2333	2.35
Unique identifiers	*issn*	0	0.00
	isbn	2	0.00
	xp	2	0.00
Special	*" - "*	58634	59.18
Month names:	*jan*	3714	3.75
	feb	3902	3.94
	mar	4575	4.62
	apr	4072	4.11
	may	186	0.19
	jun	5221	5.27
	july	4374	4.41
	aug	3800	3.84
	sep	4069	4.11
	oct	4188	4.23
	nov	3887	3.92
	dec	3927	3.96

Table 4-B The overview of records affected by Pre-Cleaning or Cleaning steps

Easy Labels:	Extraction rate	%
month_date	38037	38.39
tentative_easy_year	90435	91.28
date_american	590	0.60
date_european	910	0.92
date_japan	3667	3.70
easy_pages	41886	42.28
easy_volume	28586	28.85
easy_no	25727	25.97
easy_xp	5651	5.70
easy_issn	2193	2.21
easy_isbn	580	0.59
easy_appln_no	7425	7.49
easy_bibliographic_type	30097	30.38
easy_aetal	39280	39.65
useless	7321	7.39
useless2	11106	11.21

Table 4-C Easy Labels extraction rate

Format Labels	Extraction rate	%
nameA	13785	13.91
nameA1	4209	4.25
nameB	35002	35.33
nameB1	22939	23.15
nameC	39842	40.21
nameC1	349	0.35
nameD	40278	40.65
nameE	61531	62.11
Evaluated labels	**Extraction rate**	**%**
d_day	23773	24.00
d_month	39019	39.38
d_year	90581	91.43
pages_start	41886	42.28
pages_end	31148	31.44

Table 4-D Format and Evaluated Labels extraction

It can be observed that the amount of harmonized records in the cleaning stage is especially high for: *pages*, *et. al* and *application* tags. Significant amount of records were also found to contain patterns specified by*: tentative_easy_year, month_date, pages, volume, issue, easy_bibliographic_type and easy_aetal* attributes. High extraction rates for format based names are not surprising since both names and other substrings (that follow name formats) are expected to be extracted.

4.3 Rule statistics

Appendix Table 6-D breaks down the amount of pairs scored by each rule in the Namespace “a” and “b”. Table 4-E specifies the total amount of pairs *above the threshold* obtained at each Namespace. The order in which rules are evaluated is “a” rules first and “b” rules next. Final threshold of 14 was set in the “b” Namespace. In the Namespace “a” the threshold was set to the minimal possible score that a Double rule pair from the Namespace “a” can achieve - i.e. 7 points. Since the minimal score of a pair obtained in Namespace “b” is 5, the pairs that exceed the threshold need to be composed of some stronger rules. At each stage “Negative rules” are also used and further punish pairs composed of non-scientific references.

Sample: 99074

Pairs	Total	Threshold	Negative pairs pass point
From Namespace a rules	2272	7	18
From Namespace b rules	5511	14	15

Table 4-E Pairs returned by each rule above their selected threshold.

4.4 Precision and Recall analysis

It is important for any information retrieval system to be assessed based on how well it does its job (i.e. how well it extracts information). To this end the concept of *relevance* is most commonly used. A relevant value of an attribute is such that describes information the attribute is supposed to describe. Irrelevant values can then be viewed as mistakes of the algorithm to extract required information .

Precision and Recall are two important ratios that use information about relevance of extracted values to estimate success of the information retrieval procedure (Creighton University):

1. Precision is defined as *“ the ratio of the number of relevant records retrieved (R) to the total number of irrelevant and relevant records retrieved (I+R)*”:

$$Precision = \frac{R}{I+R}$$

2. Recall is defined as “*the ratio of the number of relevant records retrieved (R) to the total number of relevant records in the database ($R^{'}$)”:*

$$Recall = \frac{R}{R'}$$

The sum of relevant and irrelevant retrieved records for a given attribute (I+R) is easy to find, since it is the total amount of records for which a value of given attribute was extracted. In order to know how many retrieved records were relevant, one needs to compare them with a resource that specifies all relevant information for such a database. Such a resource, naturally, does not exist and it is the reason information retrieval projects are performed. However, one can estimate the success of the employed procedure by testing it on a small sample of the original database, for which a manual (or at least reliable) verification of all relevant information is performed. Such a sample is called the Golden Set.

More often than not (if the verification of the Golden set is not performed manually) the way in which the Golden Set is obtained is in fact a product of an extraction procedure. Such an extraction is subject to its own mistakes and cannot be said to obtain *all* the relevant information that a Golden Set is supposed to contain. The way such Golden Sets can nevertheless be used is by means of comparison, whereby the extraction method used in the Golden Set is considered a standard, commonly available technique, while the evaluated procedure is hoping to improve on the performance of such a standard.

The "Golden Set" used to perform precision and recall analysis of our procedure is based on the Web of Science (WOS) database. WOS is a verified database of scientific publications with reputation for reliability and large coverage[24]. WOS does not contain all scientific publications and, as a result, cannot be considered a perfect Golden Set. We however use WOS to compare efficiency of our method.

An important note is that the sample of records on which the WOS (and our) extraction procedure was performed was generated by an outsider[25].

The sample of 1276 records is focused on scientific publications (virtually all records are scientific papers) and, as a result, cannot be considered a random sample of the TLS214. For the purpose of evaluation of extraction efficiency such a sample is, however, preferable over other options, since all records can be assumed to contain relevant information. For the purpose of evaluation of clustering procedure, however, a sample would need to be closer to being random. This is to account for the fact that the procedure was designed to perform disambiguation on the whole dataset (through e.g. use of "useless" tags), rather than only on a sample of scientific records.

The Golden Set contains four attributes that overlap with the ones attempted to be extracted in our procedure. Those are: *Title*, *Author* and *Year* information. *Title* information is proxied in our method

[24] http://wokinfo.com/

[25] We would like to give your thanks to Jos Winnink of the Patent Statistics Office of the Dutch Central Planning Bureau and CWTS (Center for Science and Technology Studies) for providing us with this dataset.

by the field that contains alphabetic characters of the *residual* attribute. The idea is that once all other information is extracted from the bibliography the remaining characters (stored in *residual*) represent the title of the publication. *Author* information is provided by two attributes: Format based name and the label based on the *et. al* tag. Finally, *d_year* label is a direct counterpart to the WOS extracted Y*ear*.

Golden Set based attributes and the evaluated attributes are then compared with each other according to a most suitable metric. *Year* information was compared directly since numeric information is well suited for direct comparisons. For *name* and *title* fields the Levenshtein distance was used to approximate the most suitable threshold below which the extracted values should be considered irrelevant. Two such thresholds were used based on observation of the dataset. The table below presents the appropriate values:

Total Sample: 1276					**Levenshtein distance Above**		**Exact match**
Evaluated attribute:	**Count:**	**WOS attribute:**	**Count:**	**Evaluated attribute vs. Golden set attribute**	**0,4**	**0,45**	**-**
alphabetic residual (title proxy)	1210	*title*	651	alphabetic residual and WOS title	-	322	-
format name	949	*author*	651	format name to WOS author	364	-	-
aetal	615	*author*	651	aetal to WOS author	249	-	-
d_year	1202	*year*	652	extracted year to WOS year	-	-	648

Table 4-F Count of records considered relevant by the used Golden Set

Evaluated attribute vs. Golden set attribute	**Precision**	**Recall**	**Extraction rate WOS**	**Extraction rate of the procedure**
alphabetic residual and WOS title	0.266	0.495	0.510	0.948
format name to WOS author	0.384	0.559	0.510	0.744
aetal to WOS author	0.405	0.382	0.510	0.482
extracted year to WOS year	0.539	0.994	0.511	0.942

Table 4-G

First observation is that our method has overall higher extraction rate than use of the WOS method. With the whole sample being virtually composed of scientific records this is likely caused by the better ability of our method to find relevant information, rather than its propensity to extract irrelevant data. However, the significantly higher value of the format based name extraction rate can be attributed to the tendency of this label to extract strings which are not names, but only follow format in which names are written. High value of extraction rate for the alphabetic residual (title proxy) is attributed to the fact that almost all bibliographies contain some un-extracted information. As a result, the label is only a *proxy* for the title information. Our method of date extraction is however considerably superior to the use of WOS database, with near perfect coverage of all dates contained in the bibliographies.

A second comment is that since the extraction rate of the WOS method is low, there are a lot of attribute values that are not possible to be verified if they are relevant or not (e.g. 949-651=298 format names). This is because to asses if a record is relevant it needs to be compared with the Golden Set. The Golden set in our case does not contain all relevant records in the sample, what makes the comparison impossible. This underestimates the precision estimate of our method.

Third, the use of Levenshtein distance to asses if a value of an extracted label is the same as the one given by the Golden Set is not without faults. Namely, there are some attribute values below the specified Levenshtein distance threshold, that do describe the same *title* or *name* as the Golden Set (i.e. they are in fact relevant). They are, nevertheless, excluded from being counted as relevant matches, what further decreases precision estimate of our procedure.

The above reasons explain why precision and recall of *name* and *title* extraction are low. However, one can consider those numbers a success given the very straightforward method in which they were extracted. *Year* information extraction, however, proved to be very successful compared to the WOS method with an astounding 99% recall and very high extraction rate.

4.5 Cluster statistics

Once clusters are obtained it is possible to investigate how duplicates tend to be distributed across the dataset. As can be seen in the table and the graph below, the most common cluster size is two with 79% of all clusters falling into this class. There also seems to be a logarithmic relation between Cluster size and their recorded frequency in the final pairs set.

Amount of duplicates per cluster	Amount of clusters	%
17	1	0,05
18	1	0,05
24	1	0,05
26	1	0,05
28	1	0,05
31	1	0,05
67	1	0,05
10	3	0,14
13	3	0,14
7	6	0,27
11	6	0,27
9	**7**	0,32
8	11	0,50
6	33	1,49
5	39	1,76
4	101	4,57
3	254	11,49
2	1740	78,73
TOTAL of Multiple clusters	**2210**	**100,00**
1	90130	-

Table 4-H Amount of single and multiple size clusters from Namespace "a" and "b"

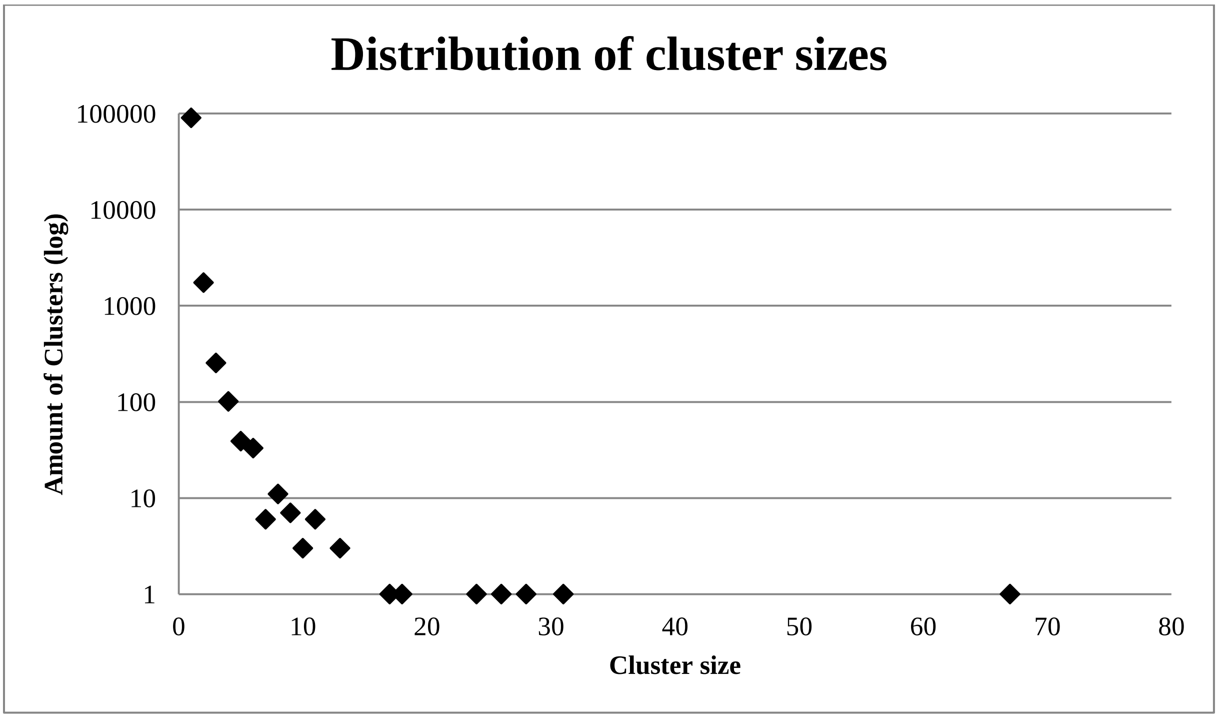


Figure 4-A Graph of the size distribution of clusters from the Namespace "a" only pairs

4.6 Visual inspection of clusters

What follows is a brief analysis of some examples of the produced clusters. This section illustrates the ability of our procedure to aggregate duplicates of the same bibliographic entity.

4.6.1 Example of a successful large cluster

Table 4-I presents a multiple record cluster that correctly aggregated the same bibliographic entity based on author *name* and *title*. *Year* information is, however, different. This example provides evidence that the procedure is able to correctly identify an entity through analysis of author information and description her publication.

cluster	new_id	npl_biblio
4	5206	Ausubel F.M. et al., 1987, Current Protocols in Molecular Biology, Greene Publishing Associates, Mutagenesis of Cloned DNA.
4	5207	Ausubel F.M., et al., 1987, Current Protocols in Molecular Biology, Greene Publishing Associates, USA; Construction of Recombinant DNA Libraries.
4	5192	Ausubel et al., Current Protocols in Molecular Biology, Greene Publishing Association and Wiley-Interscience [1987].
4	5189	Ausubel et al. Current Protocols in Molecular Biology, N.Y.:Green Publishing Associates and Wiley Interscience (1989).
4	5201	AUSUBEL ET AL.: 'Current Protocols in Molecular Biology', 1987, GREENE/WILEY
4	25279	F.M. Ausubel et. al., Current Protocols in Molecular Biology, 1987, Greene Publishing Assoc. and Wiley-Interscience, NY. (Book Not Included).
4	5122	AUSEBEL ET AL.: 'Current Protocols in Molecular Biology', 1987, GREENE & WILEY
4	5202	AUSUBEL ET AL.: 'Current Protocols in Molecular Biology', 1989, JOHN WILEY AND SONS
4	5198	AUSUBEL ET AL.: 'Current Protocols in Molecular Biolog', 1987, JOHN WILEY AND SONS
4	79	(F.M. AUSUBEL ET AL.: 'Current Protocols in Molecular Biology', 1987
4	5200	AUSUBEL ET AL.: 'Current Protocols in Molecular Biology', 1987, GREENE PUBLISHING ASSOC. AND WILEY INTERSCIENCE
4	5199	AUSUBEL ET AL.: 'Current Protocols in Molecular Biology', 1987, GREENE PUBLISHING AND WILEY-INTERSCIENCE

Table 4-I

4.6.2 Example of a correct double cluster

Table 4-J presents a typical double cluster that matches almost identical bibliography.

cluster	new_id	npl_biblio
1635	72750	Schafer et al., Recommender Systems in E-Commerce, 1999, ACM, Proceedings of the 1st ACM conference on Electronic Commerce, pp. 158-166.
1635	72749	Schafer et al., Recommender systems in E-Commerce, 1999, ACM, Proceedings 1st ACM conference on Electronic Commerce, pp. 158-166.

Table 4-J

4.6.3 Example of an imperfect double cluster

Table 4-K shows a cluster that may be incorrect since it paired publications with different year information.

cluster	new_id	npl_biblio
955	72184	SAMBROOK ET AL.: 'Molecular Cloning, A Laboratory Manual', 1989, COLD SPRING HARBOR LABORATORY PRESS
955	72193	SAMBROOK ET AL.: 'Molecular Cloning: a laboratory manual', 2001, COLD SPRING HARBOUR LABORATORY PRESS

955	72184	SAMBROOK ET AL.: 'Molecular Cloning, a Laboratory Manual', 1989, COLD SPRING HARBOR LABORATORY PRESS
955	72184	SAMBROOK ET AL.: 'Molecular cloning, A laboratory Manual', 1989, COLD SPRING HARBOR LABORATORY PRESS
955	72184	SAMBROOK ET AL.: 'Molecular Cloning, A Laboratory Manual', 1989, COLD SPRING HARBOR LABORATORY PRESS
955	72213	SAMBROOK, RUSSELL: 'Molecular Cloning: a Laboratory Manual', 2001, COLD SPRING HARBOR LABORATORY PRESS
955	72189	SAMBROOK ET AL.: 'Molecular Cloning: A Laboratory Manual', 1989, COLD SPRING HARBOR LABORATORY PRESS
955	72216	SAMBROOK; RUSSELL: 'Molecular Cloning: A Laboratory Manual', 2001, COLD SPRING HARBOR LABORATORY PRESS

Table 4-K

4.6.4 Example of a imperfect multiple entries cluster

The cluster in Table 4-M recognized some overlapping features of the records like *month* and *year*, while a slight change in description of the bibliography indicates that the records describe different entities.

cluster	new_id	npl_biblio
1059	88494	Trade Literature describing FG Products Bulkhead Systems believed to have been offered for sale prior to Jul. 20, 2001.
1059	88496	Trade Literature describing LOAD-LOK Cargo Restraint Systems believed to have been offered for sale prior to Jul. 20, 2001.
1059	88499	Trade Literature describing Schmitz Cargobull Bulkhead Systems believed to have been offered to sale prior prior to Jul. 20, 2001.

Table 4-L

4.7 Comparison of extracted XP number to the one specified in TLS214

Documentation of TLS214 specifies that the Non Patent Literature Unique Identifier (XP number) is used as the unique identifier of the TLS214 table - *npl_publ_id* – if it is detected in the bibliography field. In the investigated sample of 102,440 records:

- 5689 records incorporated XP number in the *npl_publn_id* attribute (i.e. had *npl_publn_id* < 950,000,001 what means PATSTAT detected the number in the bibliography).
- Our method extracted 5721 records with XP numbers of which 5689 had a corresponding XP number assigned by the method employed by PATSTAT. As such, there were 32 records for which PATSTAT did not (correctly or mistakenly) assign the XP number into the *npl_publn_id* attribute. From the table attached in the Appendix Table 6-F it can be seen that (overall) not assigning the XP number to the *npl_publn_id* was a mistake.
- There are also 15 records for which the extracted XP number, and the number assigned by PATSTAT did not match (see Appendix Table 6-G). This might be a reason to consider the extraction method provided by PATSTAT as less reliable.

5 CONCLUSION AND EVALUATION

5.1 Summary

The research question of this paper is: “*How to disambiguate scientific references in the PATSTAT database for the purpose of economic research and policy evaluation?*”. The solution to the problem presented in the RQ was, first, to clean and harmonize the dataset in order to obtain reliable labels that can be used for extraction of bibliographic properties of a record. Then, by looking at the regularities with respect to how certain information was transcribed we extracted attributes that were based on substrings’ format. Next, the extracted attributes were harmonized and unified to form a set of evaluated labels that accurately describe a bibliographic record. Those labels were used to construct rules that specified conditions for obtaining pairs of records that are similar in some respect. A scoring system was applied to exclude pairs, for which there was not enough evidence to consider them as true duplicates. Finally, a clustering algorithm generated clusters that represent group of duplicates of the same bibliographic entities. Construction of such sets solves the problem of record disambiguation, since the tuples which describe the same bibliographic entities are grouped together.

Based on the amount of duplicates detected in the Pre-Cleaning stage, as well as by the pairing rules we estimate that the amount of duplicate records in the TLS214 table is approximately 8%.

The obtained result facilitates economic research by providing a tool to accurately estimate use of a specific bibliographic entity in patent applications or publications. The possible areas of investigation that can use this tool include: research subsidies evaluation, university rankings[26], higher education subsidies evaluation, economics of innovation, development economics or industry specific research.

5.2 SWOT Analysis

5.2.1 Strengths

a. Simple cleaning procedures performed in the Pre-Cleaning stage reduce the amount of records that the further steps need to process. This move significantly improves the computational efficiency of the procedure, especially when working on large datasets.

b. Harmonization of records in the Cleaning stage makes extraction of patterns in the Label Extraction step more concise and transparent. Moreover, context aware pattern detection prevents erroneous conversions of substrings.

c. Implementation of Regular Expressions in the Label Extraction stage provides a flexible, efficient and extensible system for pattern extraction. Also, format based attributes were possible to be extracted with this method.

[26] See http://cwur.org/. - Center for World University Rankings. One of the used criteria is “the number of international patent filings [5%]”. This measure can be expanded to track the use of scientific publications rather than track how many patents are filed by a specific institution or associated entities.

d. Comprehensive rule formation and scoring system utilizes multitude of labels with varied degrees of reliability and coverage. This allows to test many records for being duplicates, while making sure not to produce false positive pairs.

5.2.2 Weaknesses

a. The extent to which Pre-Cleaning steps are comprehensive is unknown. More exploratory data steps should be performed to identify the most common sources of irrelevant variation that can be accounted for early on.

b. The same reasoning applies to the Cleaning stage whereby the performed harmonization was almost exclusively oriented at the labels that were extracted in the Label Extraction stage. As TLS214 contains different types of records, it is very likely the method does not provide appropriate level of harmonization for records that does not describe scientific publications.

c. The way in which some records were attempted to be excluded from being paired (the “Negative rule”) should be considered provisional. There is no proof that good majority of non scientific records was excluded from analysis in this way. Nor is it certain that some valid scientific records were not mistakenly punished, when they did contain such a label.

d. Moreover, the adapted approach attempted to apply the negative score to all the pairs that were generated by the Negative rule. Such an approach was very inefficient, as it is sufficient to apply the negative scores only to those pairs generated by other rules (i.e. those pairs that actually have a chance of going over the threshold).

e. Use of Levenshtein distance in rules is both computationally demanding and provides only an estimate for the fact that a record describes the same information. Other, more advanced, but accurate string distance measures may be more useful for this purpose.

f. There is no systematic way in which: rule score values, Levenshtein distance thresholds and the total pair score threshold (used to reject pairs) were selected. They were based on observation of behavior of data and our best judgment.

5.2.3 Opportunities

a. The way Cleaning Patters and Label Extraction codebase was implemented allows to easily extend the procedure by harmonization of further patterns and extraction of other labels. In fact some regular expressions for other, not used labels were constructed and can be used if they can prove themselves useful.

b. Pair scores were used only in so far as to reject some pairs whose score was below a selected threshold value. This method, in a sense, makes all the pairs above the threshold equally important, since the pair score is not used. An alternative method could use all the score values to create large clusters and then prune them to exclude those branches for which there is not enough evidence to support their membership in a large cluster.

c. As a result, Connected Components algorithm can be seen as a analyzed form of obtained pairs, rather than a new step that adds new information to the extraction procedure. In this respect, other clustering algorithms can prove more useful.

d. Format Based Name Extraction model provides good theoretical background for the way people transcribe names in large collections of unstructured bibliographic entries. Empirical verification of this model remains however problematic, and the extent to which names we extract are relevant, (based on the schema they are supposed to follow) is uncertain. The performed precision and recall analysis is inconclusive for this issue, because we only compare the attributes, rather than verify their content.

5.2.4 Threats

a. It is possible that harmonization and cleaning procedures in the Cleaning stage change significant information where they are not supposed to, what can contribute to overall lower *accuracy* of the method (i.e. overall correct extraction rate).

b. Format based extraction is especially prone to extracting irrelevant information what can be used as a criticism of this method to extract author names.

c. The choice of thresholds and scores was based on the sample datasets, which may not be optimal when investigating the complete dataset.

d. It is possible performance of the written algorithms and procedures is not satisfactory to be performed on the whole database. We accept this criticism since the efficiency of computational operations was not the focus of this research.

e. As discussed in the Method Outline section, our disambiguation procedure is fully deterministic – i.e. we do not account for the fact that membership of some records in a cluster is less probable than others or that a record can belong (with certain probability) to many clusters. In fact, for the problem domain in the like of TLS214, we believe that probabilistic disambiguation is a more appropriate method.

f. The Golden Set used to evaluate the extraction was incomplete and therefore the provided estimates of extraction quality are largely inaccurate. The procedure also extracted many labels that could not be evaluated because they were not part of the Golden set.

g. Precision and recall analysis was not performed on the clusters (duplicate sets) specified by the Golden Set. As such, the precision and recall of the disambiguation process is unknown.

5.3 Recommendation for future research

Multiple approaches to disambiguation of records in the TLS214 table are possible. It must be remarked that no matter their sophistication such approaches will always be less accurate than a complete solution to the disambiguation problem, which is to disambiguate records and extract their constituent information as it is added to the database. Such a solution needs to be introduced by EPO, by enforcing a consistent format of transcription of bibliographic information, as well as proper labeling of what type of information is added. We, however, believe that such a solution is likely too costly to be implemented in practice. As a result, disambiguation projects, like the one presented in this paper, are an alternative solution, before a systematic approach can be introduced.

In the meantime we are excited to see literature that uses more context aware pattern detection and harmonization. Also we believe that use of machine learning algorithms that obtain *author names*, *journal names* or even *titles* can greatly improve the quality of any disambiguation approach. We are also interested to see work into classification of documents into discrete classes for which individual, case built techniques can be applied.

Future research can also apply our methods to the complete TLS214 table of PATSTAT. Comprehensive Precision and Recall analysis of our disambiguation procedure can be conducted to verify effectiveness of the produced clusters. The framework for rule construction can also be expanded with new atomic rules and some of the composite that rules we use, but are ineffective can be excluded (for example, N2 rule seems not that useful). Such a measure can improve the parsimony of our approach. Finally, we would be excited to see the use of our procedure for the purpose of economic research or policy evaluation.

6 APPENDIX

6.1 Software and Technology

6.1.1 SQL, Transact-SQL, Microsoft SQL Server 2012 and Microsoft Management Studio

Structured Query Language (SQL) is a standardized programming language for a Relational Database Management System (Silberschatz, Korth, & Sudarshan, 2006, p. 57). PATSTAT is a relational database and its infrastructure is based on SQL concepts. Despite the fact that the paper investigates only one table of PATSTAT, the syntax of the language allows for effective manipulation of large amount of *tuples*[27]. Throughout the project many auxiliary tables are produced to store intermediate products of transformations. The relational capabilities of SQL to "link" records across those temporary tables make it well suited for this project.

Since SQL is a standard language in practice one uses its specific implementation (or so called "flavor"). The flavor used in this project is called Transact-SQL (T-SQL) and is a product of Microsoft Corp.

IDE[28] software that allows to execute T-SQL queries is called Management Studio, which is a part of 2012 Edition of Microsoft SQL Server product line. As such, the code produced in this project is verified to be compatible with the 2012 version of SQL Server package, but any compatibility with future editions of this software may not be assured.

6.1.2 C#, Common Language Runtime (CLR) and Microsoft Visual Studio 2015 RC

Another essential part of the used technology is the C# programming language. Certain advanced string manipulation techniques (See 3.1.3) are not available in the T-SQL environment. C# is used to pass input from the database, perform operations on the data using programs written in C# and then return the output back to the database environment. The way that data can be linked between the database and C# environment is possible due to Common Language Runtime (CLR) technology developed by Microsoft as part of the .NET framework. Microsoft Visual Studio 2015 RC (IDE for C#) was used to write and assemble code written in C#.

[27] A tuple can be thought of as a entry (row) of a table

[28] Integrated Development Environment - IDE

	Name	Output
RegExp Functions	*GetGroups*	Returns a capturing group of a Regular Expression
	GetMatchesCount	Returns amount of matching substring in the input string
	GetMatchesCSV	Returns comma separated matches from the input string
	IsMatch	Boolean condition for a match in an input string
	IsMatchesIndex	Returns index position of a specified match in the input string
	IsMatchesLength	Returns length of a specified match in the input string
	IsMatchesValue	Returns value of a specified match in the input string
	IsMatchIndex	Returns index position of the first match in the input string
	IsMatchIndex	Returns index position of the first match in the input string
	IsMatchLength	Returns length of the first match in the input string
	RegexReplace	Return the input string without the specified pattern
Other	*RemoveDiacritics*	Returns the input string without accents
	ComputeDistancePerc	Returns percentage similarity of strings based on Levenshtein distance
	SumIntDigits	Returns sum of all digits in a string

Figure 6-A List of C# programs that were used as functions in T-SQL

The code presented below show the use of a built-in Regex Class (*Regex*) and one of its methods: *Replace*.

```
  [Microsoft.SqlServer.Server.SqlFunction]
    public static string RegexReplace(string input, string pattern, string
replacement)
    {
        RegexOptions options = RegexOptions.CultureInvariant | RegexOptions.Compiled;
        return Regex.Replace(input, pattern, replacement, options);
    }
```

6.1.3 Regular Expressions (RegExps) and RegexBuddy 4

From the outcome of EDA it was clear that records exhibit large amount of variation with respect to transcription of similar information. Conventional solutions to normalization (or harmonization) of string data involve enumeration of all possible string variants and their conversion to a common character. This approach is however not feasible when data exerts large variation (as is the case with unsupervised human input data). Some substrings, however, can be said to follow *some* structural patterns, since most of the records do follow a convention (albeit unknown convention), in which a particular resource is described.

Prevalence of those structural patterns (as seen in Table 2-C) made the Regular Expression technology a good choice to parse strings for the given data formats. A Regular Expression is a string of special characters and literals that describes the format and content of a matching string. If a specific part of the input supplied to the RegExp engine follows a format specified by the regular expression it is returned by the RegExp engine as a match. The focus on format, rather than actual content, allows for great flexibility in specifying what sort of substrings are matched.

Regular Expressions are supported by C# through a large spectrum of pre-defined methods and classes. As a result, C# code is used to apply Regular Expression, while majority of tuple manipulation techniques are performed in T-SQL.

RegexBuddy 4[29] was not necessary to use RegExps in C#, however, it was extensively used to develop and test Regular Expressions. Moreover, few standard RegExps used in the project were derived or part of standard RegexBuddy 4, built-in library. This allowed to save time by not constructing those RegExps from scratch.

The string of characters below shows an example of one of the used regular expressions, in particular, the regular expression that extracts pages information:

```
((?<=(\bpages(\.|,)?\s?))(\d+)((?:(\s(?:to\s)?|-|/))?)(\d*))|(((\b\d+(\s(to\s)?|-
|/))?(\d+))(?=\s?pages))
```

6.2 Format Based Name Extraction Model

Model outline:

1. Declare *“$NAME$”* as a namespace.
2. $NAME$ contains two classes of objects: First name (F) and Second name (S).
3. Arity[30] of each class is limited to three elements: max $A(F) = 3$ and max $A(S) = 3$. This heuristic is given by the assumption that overwhelming majority of the population does not have (or use) names longer than 3 first names and 3 last names.
4. If F and S are combined together the arity of the full name is given by:
$$A(F,S) = A(F) + A(S).$$
5. The “natural” ordering in which names are presented is F first, S second, however, inverse order is also possible: S,F.
6. F and S are further allowed to be transcribed in two forms: *“n”* and *“N”*. “N“ stands for a word written using all capital characters, for example, “ SMITH ”. “n” is a word written with first letter capitalized and the rest written in lowercase, for example “ Smith “.
7. $[nN]$ - specifies that a word follows either “n” pattern or “N” format.
8. Further, it is allowed for F and S to be transcribed as an initial – *“I”*. An initial is a single capital character, that may or may not be dot separated.

This initial setup allows to specify 13 schemas that a person who follows the NAME namespace can use to transcribe author names. A schema is defined by specifying what class (F and S) is allowed to be assigned a particular form of transcription ($[nN]$ and I).

[29] http://www.regexbuddy.com/

[30] The amount of elements a set can contain.

Basic schemas:

1. $[nN]: F$ & $I: F$. That means words and initials are only allowed to describe first names.
2. $[nN]: F$ & $I: S$.
3. $[nN]: S$ & $I: F$.
4. $[nN]: S$ & $I: S$.

There are further 5 non basic schemas:

1. $[nN]: F, S$ & $I: F, S$
2. $[nN]: F, S$ & $I: F$
3. $[nN]: F, S$ & $I: S$
4. $[nN]: F$ & $I: F, S$
5. $[nN]: S$ & $I: F, S$

There are 4 trivial schemas:

1. $[nN]: NULL$ & $I: F, S$
2. $[nN]: NULL$ & $I: F$
3. $[nN]: F, S$ & $I: NULL$
4. $[nN]: S$ & $I: NULL$

It is obvious that some of those schemas are more useful than others. A set of stylized facts can be devised that describes principle that people can be said apply in order to evaluate and select their preferred schemas:

The optimization condition:

0. *For a given arity of a name minimize length of the NAME and maximize it being specific (accurate).*

NAME properties:

1. $NAME$ with higher arity is more specific (accurate)[31]:

$$A: 6 > A: 5 > A: 4 > A: 3 > A: 2 > A: 1.$$

2. $NAME$ that uses both classes: F and S is more specific (accurate).
3. The order of presenting NAME elements reflects accuracy:

   Order F: $F1 > F2 > F3$

[31] „>" sign is used to indicate ordering of more accurate names.

Order S: $S1 > S2 > S3$

4. If arity is the same, in general S is more specific to F: $S > F$, but it is possible that:

$$1st\ F > 3rd\ S$$

5. In general $[nN] > I$, but it is possible that:

$$\{A\!: 1\ of\ F\ class, I\ type\ name\}\ >\ \{A\!: 3\ of\ S\ class, [nN]\ type\ name\}$$

If arity of names is the same, prefer $[nN]$ centric names. E.g. with $A\!: 4$

$$A[nN]\!: AI\ \ => \ 3\!: 1 > 2\!: 2 > 1\!: 3$$

6. I is strictly preferred as homogenous. That means, although possible, "Patric A. Moris W.P." is not allowed.
7. Information cannot be repeated. This means "<u>P</u>atric <u>A</u>dam <u>C</u>hristopher P.A.C." is not allowed.

6.3 Tables

Type:	Harmonized form	Allowed Forms	Context sensitive	Symmetric
Source information	pages	p	Y	N
		pp	Y	Y
		pgs	Y	Y
		page	Y	Y
		pages	Y	Y
		seite	Y	Y
		seiten	Y	Y
	vol	v	Y	Y
		vol	Y	Y
		volume	Y	Y
		volumen	Y	Y
		volumes	Y	Y
		b	Y	Y
		bd	Y	Y
		tome	Y	Y
	no	n	Y	N
		no	Y	N
		nr	Y	N
		heft	Y	N
Author	et. al	et al	N	N
		etal	N	N
Bibliographic type	proc	proc	N	N
		proceedings	N	N
	science	sci	N	N
		wissenschaft	N	N
	chem	chem	N	N
		chemical	N	N
	natl	nat	N	N
		natl	N	N
		national	N	N
		application	N	N
	artl	art	N	N
		artl	N	N
		article	N	N
	abstract	abstr	N	N
		abstract	N	N
	magazine	mag	N	N
		magazine	N	N
	jour	jour	N	N
		journal	N	N
Negative labels	appln	appl	N	N
		appln	N	N
	pct	pct	N	N
Special	"-"	" - "	N	N

Months*	jan	jan	N	N
		january	N	N
		januer	N	N
		januier	N	N
	feb	feb	N	N
		february	N	N
		februar	N	N
		février	N	N
	mar	mar	N	N
		march	N	N
		märz	N	N
		mars	N	N
	apr	apr	N	N
		april	N	N
		avril	N	N
	may	mai	N	N
	jun	jun	N	N
		june	N	N
		juni	N	N
		juin	N	N
	jul	jul	N	N
		july	N	N
		juli	N	N
		juilliet	N	N
	aug	aug	N	N
		august	N	N
		augustus	N	N
		août	N	N
	sep	sep	N	N
		sept	N	N
		september	N	N
	oct	oct	N	N
		october	N	N
		oktober	N	N
		octobre	N	N
	nov	nov	N	N
		november	N	N
		novembre	N	N
	dec	dec	N	N
		december	N	N
		dezember	N	N
		décembre	N	N

Table 6-A. *Original diacritics are preserved in this overview. RegExp use characters without accents since they were removed in the Pre-Cleaning. Also punctuation marks are not taken into account in this list.

RegExp:	Label	Subject	Constraint	Inbetween characters	Outcome	Example(s)
Date	*month_date*	{Month name} labels	Followed by a number	Whitespace (optional)	Month name and number	may 14, may 2009

	tentative_easy_year	Numbers	Between 1850 and 2015	None	Four digit number	1993
	date_american	Sequence of three numbers	Between 1-12	"/" or "-" or "."	Up to 8 digit sequence of numbers	1/01/00, 01-01-1900
			Between 1-31			
			Between 1900 and 2099			
	date_european	Sequence of three numbers	Between 1-31	"/" or "-" or "."	Up to 8 digit sequence of numbers	04-11-96, 16.04.1955
			Between 1-12			
			Between 1900 and 2099			
	date_japan	Sequence of three numbers	Between 1900 and 2099	"/" or "-" or "."	Up to 8 digit sequence of numbers	87.12.7, 1900-04-24
			Between 1-12			
			Between 1-31			
Source	*easy_pages*	"pages" label	Followed or proceeded by a number or number range	"-" or " to " or "/" or Whitespace	Number or range	6, 56-59, 56 59, 4 to 6, 3/5
	easy_volume	"vol" label	Followed or proceeded by a number	Word boundary	Digit(s) or digit(s) with letters	3, 3a
	easy_no	"no" label	Followed by a number	Word boundary	Digits	3
	easy_xp	"XP" literal	Followed by a number with 4 up to 9 digits	Whitespace(s), "-" or ":"	XP literal and digits	XP00001234, XP123456, XP 123456, XP-123456
	easy_issn	"ISSN" label	Followed by a number with 7 to 8 digits	Whitespace, ":", "-", "X"	ISSN literal and digits	ISSN1234567, ISSN12345678, ISSN: 1234-567X
	easy_isbn	"ISBN" label	Followed by 10 to 13 digits	Whitespace, "-", "_", ", "X"	ISBN literal and digits	ISBN 3-13-136801-2, ISBN: 0-7803-8439-3, ISBN 0138544239
	easy_appln_no	"appln no" label	Digits or letter characters	Comman, dot, semicolon, "/"	Digits or letter characters	11/154
	easy_bibliographic_type	Harmonized forms of Bibliographic type labels	For some word boundaries	-	Harmonized forms of Bibliographic type labels	abstract, publn, chem

Tag Based:	*easy_aetal*	Detects "et. al" label and matches between 1 and 4 words to the left of the label	Escapes on special characters like brackets, slashes and other	Dot and comma	Up to four words	J Smith, M Anderson
Negative labels	*useless*	Literal "See references of "	suffix EP or WP or WO	-	Boolean value	1, 0
	useless2	Literals "&#x", "pct", "appln no", "appln serial no", "publn no", "publn serial no"	Case insensitive	-	Boolean value	1, 0

Table 6-B

RegExp:	**Label**	**Subject**	**Constraint**	**In-between characters**	**Outcome**	**Examples**
Format based:	*nameA*	[nN]{1,3}I{1,3}[nN]{1,3}	Only valid if the corresponding auxiliary match of nameA1 is not null	Whitespace, "-", "."	3 part name with variable count of terms per each part	John Adam S SMITH, John S.A. Wright-Philips
	nameA1	[nN]I[nN]	Case insensitive, but lowercase names must begin with a capital letter	Whitespace, "-", "."	3 part name with single term per part	John S. Smith, Patric J ADAMS
	nameB	I{1,3}[nN]{1,3}	Only valid if the corresponding auxiliary match of nameB1 is not null	Whitespace, "-", "."	2 part name with variable count of terms per each part	J.A.P. Smith, J. Adam SMITH
	nameB1	I[nN]	Case insensitive, but lowercase names must begin with a capital letter	Whitespace, "-", "."	2 part name with single term per part	J. Smith, J SMITH
	nameC	[nN]{1,3}I{1,3}	Only valid if the corresponding auxiliary match of nameC1 is not null	Whitespace, "-", "."	2 part name with variable count of terms per each part	Smith J.A.P., Adam SMITH J.
	nameC1	[nN]I	Case insensitive, but lowercase names must begin with a capital letter	Whitespace, "-", "."	2 part name with single term per part	Smith J., SMITH J
	nameD	[nN]{2,3}	Words have to have at least 3 characters	Whitespace	2 to 3 part name with no initials	JOHN SMITH, John Adam Smith
Ordering based:	*nameE*	Alphabetic words at the start of the string	Before first comma or colon	Dot	String of words	John Patric; Pavel Colins,
Non RegExp properties	*s_start*	Start of the string	8 characters	-	8 characters	"Smith J,"
	s_end	End of the strong	8 characters	-	8 characters	"a, May 1"
	bib_numeric	Numbers	Preserve single space between numbers	None	All digits in the original string	22 221 200 292 109 200, 26 132 003 297 303
	bib_alphabetic	Alphabetic characters	Preserve single space between words	None	String of words	[Murakawa et al Biosynthesis of DErythroascorbic Acid by Candida Agric Biol chempages], [A Rejasekar MCATA Meta Information Catalog versionmar]
	bib_alphanumeric	String of alphanumeric characters and whitespaces	Single spacing and lowercase	None	String of words and numbers	[aw dox organic synthesis vol 1 1941 pages 5], [a van der horst msc thesiseindhoven 2007]

	sum_of_num	Calculates sum of all numbers in the string	-	-	Integer	12569
	count_of_num	Calculates count of all numbers in the string	-	-	Integer	8
	npl_biblio_length	Calculates string length of the record	-	-	Integer	154

Table 6-C Format based extracted labels

		Sample	**99074**
Double rules:		**Pair Count**	**%**
	Rule A	622749146	-
N	**W**		
1	1a	252	3,07
1	1b	3838	46,77
1	2a	631	7,69
1	3a	738	8,99
1	2b	227	2,77
1	3b	78	0,95
1	4	917	11,17
1	5a		
1	5b	149	1,82
2	pages	48	0,58
2	1a		
2	1b		
2	2a	15	0,18
2	3a		
2	2b		
2	3b		
2	4		
2	5a		
2	5b		
3a	1a	13	0,16
3a	1b	22	0,27
3a	2a	33	0,40
3a	3a	7	0,09
3a	2b	50	0,61
3a	3b	12	0,15
3a	4	28	0,34
3a	5a	23	0,28
3a	5b		
3b	1a	135	1,65
3b	1b	149	1,82
3b	2a	249	3,03
3b	3a	142	1,73
3b	2b	375	4,57
3b	3b	28	0,34
4	1a		
4	1b	20	0,24
4	2a	8	0,10
4	3a	14	0,17
4	2b	5	0,06
4	3b		
	TOTAL	8206	

Table 6-D Double rules list with count of scored pairs by each rule

6.3.1 T-SQL Query of a rule

```
--N3bW2b
if object_id('rule_N3bW2b') is not null drop table rule_N3bW2b
select distinct a.new_id as new_id1, b.new_id as new_id2
into rule_N3bW2b
from evaluated_patterns as a
join evaluated_patterns as b on
       (a.pages_start=b.pages_start
       and a.volume=b.volume
       and a.d_year=b.d_year)
where a.new_id < b.new_id
       and a.aetal is not null
       and b.aetal is not null
       and dbo.ComputeDistancePerc(a.aetal, b.aetal) >= 0.70
except (select * from rule_N3aW2a)
go
```

Example of how a pair set of a N3bW2b rule is obtained. The table with evaluated attributes is self joined with each other and two set of restrictions are enforced on this join: 1) N3b (*pages_start*, *volume*, *year*) attributes need to be equal across the two records 2) Levenshtein distance between *aetal* attributes of the two records needs to be higher than 70%.

new_id	npl_publn_id	npl_biblio	xp_check
396	965413039	3GPP TS 36.300 v. 8.4.0, Release 8, Mar. 2008, AN-XP014041816; pp. 1-126.	XP014041816
3537	959484998	Anderson et al., NASA Contractor Report, (Online), No. NASA-CRz-180844, Jun. 1, 1987, (XP002453502). Washington, DC, USA, Retrieved from the Internet: URL: http://Hdl.handle.net/2060/19900001894.	XP002453502
3870	967340062	Anonymous: << AT&T breakthrough speech technology designed to increase sales land customer service productivity >>, AT&T News Release, 'Online!, Sep. 30, 2003, pp. 1-2-XP00232137.	XP00232137
8409	970472769	Blumen färben, , 19 June 2010 (2010-06-19), XP055051188, Retrieved from the Internet: URL:http://web.archive.org/web/20100618235831/http://www.wdr.de/tv/wissenmachtah/bibliothek/blumenfaerben.php5 [retrieved on 2013-01-25]	XP055051188
20242	957394553	Development and Operation of the Next-Generation Rating/Filtering System on the Internet, (XP002219058). Retrieved from the Internet:URL:http://www.nmda.or.jp/enc/rating2nd-en.html on Oct. 30, 2002.	XP002219058
27089	970472768	Francis W. Holmes: Distribution of dye in elms after trunk or root injection, Arboriculture & Urban Forestry Online, 1 September 1982 (1982-09-01), pages 250-252, XP055045440, Retrieved from the Internet: URL:http://joa.isa-arbor.com/request.asp?JournalID=1&ArticleID=1831&Type=2 [retrieved on 2012-11-23]	XP055045440
27856	962036620	G. Anteniese et al., Some Open Issues and New Directions in Group Signatures, Financial Cryptography, Third International Conference, FC '99 Proceedings, pp. 196-211, (XP002252934) Springer-Verlag, Berlin, Germany.	XP002252934
30834	968745642	Gultekin, M. et al., Styrenation of castor oil and linseed oil by macromer method, Macromol. Mater. Eng., vol. 283, 2000, pp. 15-20 (XP002522953).	XP002522953
32581	968691891	He, Xun et al., Ionic-Tag-Assisted Oligosaccharide Synthesis, Synthesis, No. 10, pp. 1645-1651, EXP-002521376, (2006).	XP-002521376
35796	952798122	IEEE TRANSACTIONS ON GEOSCIENCE AND REMOTE SENSING vol. 26, no. 6, November 1988, NEW YORK US pages 733 - 739 , XP1388 MICHIGUCHI ET AL. 'Advanced Subsurface Radar System for Imaging Buried Pipes'	XP1388
35970	955570212	Image Indexing and Retrieval Using Visual Keyword Histograms. Joo-Hwee Lim and Jesse S. Jin. 0-7803-7304, Aug. 26, 2002. vol. 1, p. 213-216. C2002 IEEE. (XP010604344).	XP010604344
36432	962934240	Integral Distributed Battery Pack for Portable Systems, Research Disclosure, No. 333, Jan. 1, 1992, p. 12 (XP000281124).	XP000281124
36478	961696653	Intel, www.intel.com/design/network/products/npfamily/ixp2800.htm, Intel IXP2800 Network Processor, Oct. 10, 2003.	XP2800
38889	962036621	J. Camenisch et al., Efficient Group Signature Schemes for Large Groups, Advances in Cryptology-Crypto '97 Proceedings of the Annual International Cryptology Conference, pp. 410-424 (XP000767547), Berlin, Germany.	XP000767547
39077	962036624	J. Kilian et al., Identity Escrow, Advances in Cryptology, 18<SUP>th </SUP>Annual International Cryptology Conference, Proc. Lecture notes in Computer Science, vol. 1462, pp. 169-185, 1998, (XP000792174) Berlin, Germany.	XP000792174
40709	962036623	Jinn-Ke Jan et al., A Secure Electronic Voting Protocol with IC Cards, Security Technology, 1995, Proceedings IEEE, pp. 259-265, (XP010196424), New York, NY.	XP010196424
46587	962036622	L.F. Cranor et al., Sensus: A Security-Conscious Electronic Polling System for the Internet, Proceedings of the 13<SUP>th </SUP>Hawaii International Conference of Wailea, HI, 1997, pp. 561-570, (XP010271743).	XP010271743

49771	955494165	M. Ahmad, et al., Ortho Ester Hydrolysis: Direct Evidence For A Three-Stage Reaction Mechanism, Engineering Information, Inc. NY, NY, vol. 101, No. 10 (XP-002322843), 1979.	XP-002322843
53413	960538882	Microsoft Corporation: Understanding SAMI 1.0 Microsoft Developers Network, (Oct. 1, 2001),XP007902747.	XP007902747
56905	954650640	Nomura et al., A Bitrate and Bandwidth Scalable Celp Coder, IEEE Int'l Conf. On Acoustics, Speech & Signal Processing, Seattle, WA May 12-15, 1998, pp.341-343,XP002112625.	XP002112625
60260	961481692	Organic Electroluminescent Device Luminescent Layer Contain Polymer Umbelliferyl PolymethacrylateXP-002224027 (1992).	XP-002224027
66174	962648499	Piumarta et al., Optimizing Direct Threaded Code by Selective Inlining, Assoiciation for Computing Machinery, vol. 33, No. 5, pp. 291-300, May 1, 1998.XP-000766278.	XP-000766278
82416	962038029	Signal Processing of HDTV L Aquila, Italy Feb. 29 Mar. 2, 1988 (XP 00075084 / pp. 471 485).	XP 00075084
85861	957394552	T. Negrino, The MacWorld Web Searcher's Companion, MacWorld, PC World Communictions, San Francisco, CA, US, vol. 17, No. 5, May 2000, pp. 76-82 (XP008019722).	XP008019722
95677	952218555	WILDING P. ET AL: 'PCR in a Silcon Microstructure' CLINICAL CHEMISTRY vol. 40, no. 9, 1994, pages 1815 - 1818, XP000444699	XP000444699
97231	955494164	Y. Chiang et al., Hydrolysis Of Ortho Esters; Further Investigation Of The Factors Which Control The Rate-Determining Step, Engineering Information, Inc. NY, NY, vol. 105, No. 23 (XP-002322842), 1983.	XP-002322842
97288	962036625	Y. Mu et al., Anonymous Secure E-Voting over a Network, Computer Security Applications Conference, 1998, Proceedings 14<SUP>th </SUP>Annual, pp. 293-299, 1998 (XP010318642), Los Alamitos, CA.	XP010318642
98005	964231672	Yoshida, Hu , LUN Security Considerations for Storage Area Networks, Hitachi Data Systems Paper-XP 002185193 (1999), 1-7.	XP 002185193
98732	957456969	Zhao, Jian et al., Embedding Robust Labels Into Images for Copyright Protection, (XP 000571967), pp. 242-257, 1995.	XP 000571967
99067	956498383	Z-World Products and Services re: XP8100, 2 pp., printed May 24, 1999.	XP8100
99068	956498384	Z-World Products and Services re: XP8500, 1 p., printed Jun. 3, 1999.	XP8500
99069	956498385	Z-World Products and Services re: XP8700, 1 p., printed May 24, 1999.	XP8700

Table 6-E XP numbers not assigned by PATSTAT

new_id	npl_publn_id	xp number	npl_biblio
26937	1956222	19562220	Forte, M. et al. Optimization of a Dielectric Barrier Discharge Actuator by Stationary and Non-Stationary Measurements of the Induced Flow Velocity: Application to Airflow Control, Experiments in Fluids; Experimental Methods and Their Application to Fluid Flow, Springer, Berlin, Germany, vol. 43, No. 6, pp. 917-928, Aug. 1, 2007, XP-019562220.
26469	807033	8070336	Fischer, et al., Electrocatalytic properties of mixed transition metal tellurides (Chevrel-phases) for oxygen reduction, Journal of Applied Electrochemistry, vol. 25, (1995) m oo 1004-1008, XP 008070336.
50166	806906	8069069	M. Waksmundzka-Hajnos, Chromatographic Separation of Nitro-Phenones and Their Reduced Derivatives on Thin Layers of Polar Adsorbents, XP-008069069, pp. 159-171.
54515	807019	8070191	Moreau et al., Synthese d'indomonocarbocyanines a elimination biliaire selective Etude experimentale chez l'animal, Eur. J. Med. Che-Chimica Therapeutica, May-Jun. 9, 1974, No. 3, pp. 274-280, XP-008070191.
97108	1970049	19700494	XP 019700494 (WEI HE et al.): ISO-PARAMETRIC CNC TOOL PATH OPTIMIZATION BASED ON ADAPTIVE GRID GENERATION; ISSN 1433-3015; The International Journal of Advanced Manufacturing Technology, SPRINGER, BERLIN, vol. 41 no. 5-6. 22-May-2008, pages 538-548.
48375	806981	8069811	Lie Ken Jie et al., Lipase Specificity Toward Some Acetylenic and Olefinic Alcohols in the Esterification of Pentanoic and Stearic Acids, Lipids, vol. 33, No. 9, pp. 861-867, XP 008069811, 1998.
11842	264666	2646666	Cationic polymeric thickeners useful in fabric softeners, Research Disclosure Database No. 429116, Jan. 1-31, 2000, p. 136, XP-002646666, ISSN: 0374-4353.
49027	914531	9145313	lkuo Hayashi et al., Generation of Monoclonal Antibodies Against the Extracellular Domain of Nicastrin, Alzheimer's & Dementia: The Journal of the Alzheimer's Association, Jul. 1, 2006, vol. 2, No. 3, Suppl. 1, P3-412, XP 009145313, p. S497.
93208	807084	8070847	Vegt et al.-Renal Uptake of Radiolabeled Octreotide in Human Subjects Is Efficiently Inhibited by Succinylated Gelatin, The Journal of Nuclear Medicine, vol. 47, No. 3, Mar. 2006, pp. 432-436, XP-008070847, ISSN: 0161-5505.
44077	118503	1185032	Kernel-Based Object Tracking, Dorin Comaniciu, Senior Member, et al., IEEE, May 5, 2003, XP-001185032.
97122	264667	2646674	XP-002646674-Space-time tradeoff-Wikipedia.
94609	806980	8069801	Warwel et al., An Efficient Method for Lipase-Catalysed Preparation of Acrylic and Methacrylic Acid Esters, Biotechnology Techniques, vol. 10, No. 4, pp. 283-286, XP 008069801, Apr. 1996.
93970	806903	8069039	W. Waiers, Some Substitution Reactions of 4-Aminodiphenylmethane, XP-008069039, pp. 1060-1064.
22807	1979769	19797696	Elizaveta Kon et al: Platelet-rich plasma: intra-articular knee injections produced favorable results on degenerative cartilage lesions , vol. 18, No. 4, Oct. 17, 2009, pp. 472-479, XP 019797696.
87853	806889	8068898	Thomas Wedi, Adaptive Interpolation Filter for Motion Compensated Hybrid Video Coding, Proceedings of the Picture Coding Symposium, Apr. 25, 2001, XP-008068898, pp. 49-52.

Table 6-F Conflicting XP numbers between PATSTAT and our method

7 BIBLIOGRAPHY

Codd, E. F. (1970). A Relational Model of Data for Large Shared Data Banks. *Communications of the ACM* (6), pp. 377-387.

Constans, P. (2009). A Simple Extraction Procedure for Bibliographical Author Field. *http://arxiv.org/abs/0902.0755* .

Creighton University. (n.d.). *Measuring Search Effectiveness* . Retrieved 07 26, 2015, from creighton.edu: https://www.creighton.edu/fileadmin/user/HSL/docs/ref/Searching_-_Recall_Precision.pdf

EPO. (2015, 4 1). Data Catalog v5.03. *DATA CATALOG for PATSTAT* , 58.

EPO. (2015, 4 8). *Non-Patent Literature reference numbers (XP)*. Retrieved 7 27, 2015, from http://worldwide.espacenet.com/: http://worldwide.espacenet.com/help?locale=en_EP&method=handleHelpTopic&topic=accessionnumber

Leydesdorff, L., & Milojević, S. (2015). Scientometrics. In M. Lynch, *International Encyclopedia of Social and Behavioral Sciences* (Vol. Science and Technology Studies). Elsevier.

Mowery, D., & Rosenberg, N. (1979). The influence of market demand upon innovation: A critical review of some recent empirical studies. *Research Policy , 8*, 102-153.

NIST/SEMATECH. (2012, April). *What is EDA?* Retrieved 7 2015, 25, from e-Handbook of Statistical Methods: http://www.itl.nist.gov/div898/handbook/eda/section1/eda11.htm

Rosen, H. S., & Gayer, T. (2008). *Public Finance* (8th ed.). McGraw-Kill.

Silberschatz, A., Korth, H. F., & Sudarshan, S. (2006). *Database System Concepts 6th Edition.* New York: McGraw-Hill.

Stanford Encyclopedia of Philosophy. (2013, May 8). *http://plato.stanford.edu/.* Retrieved July 27, 2015, from Well-Being, 4.3 Objective List Theories: http://plato.stanford.edu/entries/well-being/#ObjLisThe

Tan, P. N., Steinbach, M., & Kumar, V. (2006). *Introducton to Data Mining.* Boston: Pearson Education.

Winnink, J. (2015). *Science-technology interaction (an annotated bibliography).*

Winnink, J., & Kracker, M. (2015, Apr 24). *Multiple items in NPL_BIBLIO strings*. Retrieved 7 27, 2015, from http://forums.epo.org/: http://forums.epo.org/patstat/topic3005.html