Extraction, Exploitation and Evaluation of Document-based Knowledge

Antoine Doucet

Université de Caen Basse-Normandie
GREYC – CNRS UMR 6072
ED SIMEM

Habilitation Thesis
in Computer Science

Extraction, Exploitation and Evaluation of Document-based Knowledge

Antoine Doucet
to be defended on April 30, 2012

Committee:

Dr. Massih-Reza Amini, UPMC-LIP6,	Reviewer
Pr. Pavel Brazdil, University of Porto (Portugal),	Reviewer
Pr. Manuel Vilares Ferro, Universidade de Vigo (Spain),	Reviewer
Pr. Bruno Crémilleux, Université de Caen Basse-Normandie,	Examiner
Pr. Mounia Lalmas, Yahoo! Research Barcelona (Spain),	Examiner
Pr. Isabelle Tellier, Paris 3 Sorbonne Nouvelle,	Examiner
Pr. Gaël Dias, Université de Caen Basse-Normandie,	Advisor

Contents

Chapter 1

Introduction

My research gravitates around digital documents: knowledge extraction, knowledge exploitation, and subsequent evaluation, both from a theoretical and an applied point of view.

The genericity of the techniques employed is to the core of my works, with a particular attention given to issues of scalability. Notably, the algorithms presented are essentially applied to text, but they function for any type of sequential data.

When it comes to text, the generality and robustness of the techniques allow for endogenous applications functioning for any language, any domain, and any type of document, or collection of documents. Subsequently, experiments cover languages written with different alphabets, and belonging to different linguistic families. The methods developed may be applied at the sentence, word, or character level. The document collections range from news feed collections to scientific articles, to vast collections of digitized books.

1.1 Why Genericity?

An important specificity of this work is a focus on the development of techniques that generalize and scale. A list of practical examples follows, in different respects:

- Extraction of knowledge: text is seen as a special type of sequential data, where items can for instance be chosen to be words, or characters.

- Languages: the methods are language-independent, and do not generally require other resources than those found within the text itself. Stoplists, morphological analysis, POS tagging, are all notably out of scope.

- Structured documents: document schemas or DTDs are irrelevant and the techniques should function even if documents are not well-formed.

- Corpora: the domain is irrelevant, whether documents are specialized or not (e.g., news feeds, scientific articles or whole books dealing equally with finance, computer science or poetry).

These specificities have been constant guiding lights throughout my research, from master's degree to this habilitation thesis.

However, the ultimate goal of my research is evidently not to prove that linguistic knowledge serves no purpose in natural language processing (NLP). In the contrary, linguistic knowledge is present throughout this manuscript, with a main focus on general properties of human languages. By these means, the real objective is to reach intermediary, language-independent milestones, to be improved upon with the careful integration of language-specific techniques.

In other words, leaving language-dependent resources aside is a way to set baselines for languages for which such resources exist, and it is filling a gap for the many more languages for which resources are scarce.

In my view, this line of research actually aims at the same objective as mainstream monolingual approaches. The only difference resides in the choice of priorities and subsequent intermediary steps. Most research aims at excellence in one language first, before possibly trying to reproduce the technique for a few other languages. If the new language to be handled is radically different from the initial one, this approach often translates "adapting the system to a new language" into "building a new system".

In the contrary, my approach aims at first factorizing as much as can be, to reach as good as possible results for all languages, before aiming at excellence for a number of desired languages. This allows to postpone the decision of which languages are to be the focus of the end-user, and makes global applications possible.

From a personal point of view, it is possible that this uncommon approach of documents is the consequence of a slow landing in the area of text material, after graph theory projects in the University of Tennessee and a master's degree (D.E.A.) specialized in algorithms with a final internship focused on data mining. This probably made it more natural to me to look at text as a special case of sequential data, and to look at structured documents as trees with particularly interesting leaves.

It may also be the environment of my early research that led me to language-independent works. Perhaps I was simply not able to chose between my mother tongue (French), my professional tongue (English) and my surrounding tongues in bilingual Finland (Finnish, and to lesser extent, Swedish), and I have never accepted to let my research leave any of these personally important languages aside. This small set already contains Latin, Anglo-Saxon and Finno-Ugric languages, from the isolating and agglutinating families, which left me with little choice but to develop resource-free methods.

This environment also made me see early that Finnish, the language of 5 million people in one of the most well-off eurozone countries, was still too small to trigger the economic opportunity of numerous Finnish-specific language tools. With very many languages in a similar situation, it made it clear to me that developing language applications that would function with no external resources was not just a set of exciting puzzles to be solved for the sake of self-satisfaction, but the key to much-needed applications.

1.2 Research History and Background

During my final training as a master's student in Helsinki in 2001, I was working on the extraction of text data from semi-structured document collections. The use of structure in document description is the first topic I ever addressed in my research

career. As a direct consequence of this work, my very first publications presented a technique to extract text sequences, integrating structural information linked to their location in the XPath of structured documents [34, 35].

This work evidenced the possibility to use document structure to enrich knowledge about text descriptors. It also hinted that logical structure may sometimes be used as a substitute for semantic information.

In late 2001, following my internship as a master's student, I started a joint Ph.D. thesis (*cotutelle*) in the University of Helsinki and in the University of Caen Lower-Normandy, under the supervision of Helena Ahonen-Myka in Helsinki and Bruno Crémilleux in Caen. However, I conducted most of my doctoral research in the University of Helsinki, where I was employed as a researcher. My Ph.D. thesis focused on the extraction and selection of sequential patterns from text, and their experimental use in information retrieval. Work stemming from my doctoral work, defended in 2005, is presented in Sections 2.1 and 2.2 of the knowledge extraction chapter, and in Section 3.1 of the knowledge exploitation chapter.

My Ph.D. started with the topic "text mining", offering me a vast field to explore until I settled down to a more focused problem. Before my interest grew in sequential pattern mining and multiword units, I naturally continued investigating into document structure, as a follow-up of my my master's thesis. In actuality, even while my doctoral work was focused on sequential description, I continued spending a good share of my time investigating into the field of structured documents.

This involvement mainly took ground in 2002, with the study of the inclusion of structural data within the process of clustering XML documents [32]. To date, this technique remains a strong baseline. A one-time participant of the INEX mining track in 2006 (which did not exist in 2002), its results were in the top-tier, reaching the 1^{st} rank (Wikipedia collection, out of 7) and 2^{nd} rank (IEEE collection, out of 13) [27]. Section 3.4 describes this work on the unsupervised classification of XML documents.

My research on structured documents evolved into the field of structured information retrieval, as described in Section 3.3. The corresponding EXTIRP system, developed under my lead by a team of 6 people in 2003 [31], was continuously expanded until 2009 [14, 23, 24, 26, 28, Leh05, Leh06a, Leh06b]. EXTIRP laid ground for several research initiatives within the Doremi research group of the University of Helsinki.

Starting in 2007, my work around structured documents expanded towards evaluation methodologies and digitized books, which are the focus of the last chapter of this manuscript (Chapter 4). It describes the adaptation of existing information retrieval methodologies to massive book collections (Section 4.2), and the full set up of an evaluation methodology for the extraction of logical structure from digitized books (Section 4.3).

Since 2010, I have had the privilege to co-advise the Ph.D. thesis of Gaël Lejeune, together with Nadine Lucas. This work aims at using discourse structure for multilingual extraction (see Section 2.3). After I obtained the title of *Dosentti*[1] of the University of Helsinki in 2011, and in conjunction with my temporary position

[1]The closest Finnish equivalent to the French HDR, it notably allows one to supervise Ph.D. students after a defense and positive external reviews based on career and publications – a major difference with the French system is that there is no thesis to be written.

as a full-time CNRS researcher (*délégation*), I have started to co-advise in Helsinki with Hannu Toivonen, on the topic of the general detection of novelty from within news feed streams (this work is discussed in the conclusion of Section 2.3 as well as in the perspectives of this manuscript in Chapter 5).

1.3 Organization of this Monograph

This dissertation is organized in three main chapters, following the three main steps of the document processing pipeline, from the apprehension of documents to the evaluation of applications.

Extraction of knowledge. The first two sections of Chapter 2 deal with the core of the works of my Ph.D. thesis and follow-ups. Most of it was achieved at the University of Helsinki between 2002 and 2005 under the joint supervision of Helena Ahonen-Myka in Helsinki and Bruno Crémilleux in Caen. Section 2.1 concerns the extraction of maximal frequent sequences (MFS) from any type of sequential data, while Section 2.2 describes a statistical technique for filtering the most interesting sequences. Applying this work to text allows for the extraction of phrasal descriptors, although the generality of the technique implies that it can be applied to any type of sequential data.

Section 2.3 introduces the doctoral work of Gaël Lejeune, whom I have been co-supervising with Nadine Lucas since October 2010. The technique presented relies on discourse-based features to perform multilingual information extraction, in the application field of epidemiological events surveillance. Unlike the vast majority of state of the art methods, the technique does not rely on extraction patterns, and performs comparably to the state of the art, in the few languages that state of the art techniques are currently addressing.

Exploitation of knowledge. Chapter 3 presents applications that make use of extracted knowledge, once again preserving independence from the domain, language and type of documents in the corpus.

In Section 3.1, we present a general method to calculate the phrase-based similarity of documents, that we applied to information retrieval. The genericity of the approach is underlined by experiments in Japanese, Chinese, Korean and English, and on scientific and news feed articles.

Section 3.2 presents the outcome of a collaboration with the *Hultig* group of the University of Beira Interior (Portugal)[2], in which MFSs have proved particularly useful as a prior to allow the one-pass multiple-sequence alignment of paraphrases. The resulting alignments were then used to discover word semantic relations by observing variations of vocabulary.

The last two sections of this chapter are dealing with the exploitation of structured information. In Section 3.3, we introduce the EXTIRP structured information retrieval system, designed and developed under my lead in 2003 at the University of Helsinki, and used yearly within the INEX evaluation initiative until 2009. It

[2]Then led by Gaël Dias, who joined the University of Caen Lower-Normandy as a full professor in October 2011, and happens to be the advisor of this habilitation thesis.

addressed the problem of granularity in information retrieval by defining minimal retrieval units at lower levels of the document tree, before propagating relevance values in a bottom-up fashion, to finally decide on the optimal document fragments to be returned to a query. EXTIRP has been the framework of Miro Lehtonen's Ph.D. thesis, defended in 2006, which triggered a considerable amount of joint research and corresponding co-authored publications. The last application presented in this chapter is that of the non-supervised classification of XML documents. We proposed to rely on structural features to efficiently distinguish outliers, before performing regular clustering (Section 3.4).

Evaluation. Chapter 4 presents research on methodological aspects of evaluation. It describes work conducted in the evaluation of book retrieval and book structure extraction. The first section describes the context and motivation of this line of work, and introduces the book collection that we made available to the community (Section 4.1).

This part of my work was entirely led together with Gabriella Kazai from Microsoft Research Cambridge (United Kingdom). Within this collaboration, the repartition of tasks is essentially that she has been the leader of the book retrieval task, discussed in Section 4.2, while I have been leading the book structure extraction task, described in Section 4.3. Following their growth, both tasks were later supported by additional co-organizers.

We describe the full set up of the evaluation frameworks, from collection distribution to metrics and distribution of results, in the specific context of large collections of very large documents. The chapter is naturally focused on the research contributions in evaluation methodology and it leaves off the practical aspects of the organization of the book track.

Convention for references within this manuscript. To facilitate the identification of the contributions of the author, all personal references are cited as numbers throughout the document (e.g., [25]). They are listed in the "Personal Publications" section, page 123.

All other publications are cited with the initials of their authors followed by the corresponding year of publication (e.g., [PBMW98]), and listed in the "Bibliography" section, page 129.

Summaries of related publications. In addition, to facilitate the apprehension of the contributions specific to the different sub-topics of this manuscript, each of the three chapters contains a "Related Publications" section, where the author's work and main publications in the corresponding domain are put in context.

Chapter 2

Extracting Knowledge

The very first step of a text-based process is the extraction of information. In this part of the manuscript, we will introduce techniques developed to extract different types of information.

The techniques introduced in the first two sections (2.1 and 2.2) describe the core of the works of my Ph.D. thesis and follow-ups. Most of it was achieved at the University of Helsinki between 2002 and 2005 under the joint supervision of Helena Ahonen-Myka in Helsinki and Bruno Crémilleux in Caen, within a joint thesis agreement (*cotutelle*). It concerns the extraction of maximal frequent sequences (MFS) from any type of sequential data. The approach relies on two main steps. The first step, described in Section 2.1 is the extraction of sequences in a resource-free and unsupervised fashion. The second step is the statistical filtering of the most interesting sequences (see Section 2.2).

The wide spectrum of the techniques defined means that when the method is applied to text, as was the case in my Ph.D. thesis, it is by nature fully independent of the domain and language of the corpus at hand. Furthermore, the granularity employed is free to choose: it is possible to extract sequences of characters from a set a scientific articles, or sequences of words from a set of sentences. In the case of sequences of words, many of the extracted sequences are multiword units (MWU).

Section 2.3 deals with multilingual information extraction. The corresponding work stems from the Ph.D. work of Gaël Lejeune, whom I have been supervising together with Nadine Lucas since October 2010. The main application is the detection of epidemiological events from online news feeds. The essential specificity of the technique is the ability to detect events efficiently, with short lexicons as the only external resources (a few hundred words). Unlike main state of the art techniques, we do not rely on structural patterns during the extraction process. The subsequently strong reduction in external resources implies that the technique is both very efficient and very easily adapted to corpora written in different languages and concerning different domains. Up to this point, however, the work has essentially focused on demonstrating its capacity to detect events written in distinct languages. This represents important progress in a field where detecting new information as soon as possible is crucial, and it is a matter of fact that new information is usually first described in the language of the region where it occurs. Its occasional translation into a major language is a sign that the importance of the piece of information was already acknowledged. Hence, while waiting for translation into a (major) language may allow to detect events, it is by essence failing to detect *new*

events.

This is why, in our view, a useful surveillance system shall not be limited in the number of languages that it can deal with. This is especially true for applications in which high recall is important (e.g., epidemiology, finance, ...).

2.1 Extracting Sequential Data from Text

This section is focused on the extraction of frequent patterns from sequential data. Specifically, it deals with the extraction of Maximal Frequent Sequences (MFS), within the application framework of textual data. This work stems from my doctoral work as well as more recent follow-ups.

2.1.1 Extraction of Multiword Units

When MFSs are extracted from text, many of the resulting word sequences are multiword units. Due to the higher information content and specificity of phrases versus words, researchers have always been interested in multiword units (MWU) for information access. However, their extraction is difficult.

The first extraction models, introduced until the late 1980's, came with numerous restrictions. Mitra et al. [MBSC87], for example, defined phrases as adjacent pairs of words occurring in at least 25 documents of the TREC-1 collection. Choueka et al. [CKN83] later extracted adjacent word sequences of length up to 6. The extraction of sequences of longer size was then intractable. The adjacency constraint is regrettable, as natural language often permits to express similar concepts by introducing one or more words between two others. For example, the phrases "President John Kennedy" and "President Kennedy" are likely to refer to the same person.

A new trend started in the 1980's, when linguistic information began to be used to filter out "undesirable" patterns. The idea consists in using parts-of-speech (POS) analysis to automatically select (or skip) the phrases matching a given set of linguistic patterns. The "classics" of extraction techniques do rely on a combination of statistical and syntactical methods [Sma93, FAT98].

However, the use of POS tagging is a serious restriction at a time when multilingual information retrieval remains an important trend. Therefore, we think it is of crucial importance to propose language-independent techniques for MWU extraction, but there is surprisingly very few research in this direction, as was suggested by a recent workshop on multiword expressions [KRV11] where most of the 10 accepted papers presented monolingual techniques, designed for a total of 6 distinct languages (German, English, Korean, Bengali, Basque and French).

One of the few methods that do not require language resources was introduced by Dias et al. [Dia03, DGBPL00], in an elegant generalization of conditional probabilities to n-grams extraction. The normalized expectation of an n-words sequence is the average expectation to see one of the words occur in a position, given the position of occurrence of all the others. Their main metric, the mutual expectation, is a variation of the normalized expectation that rewards n-grams occurring more frequently. While the method is language-independent and does not require word adjacency, it still recognizes phrases as a very rigid concept. The relative word positions are fixed, and to recall our previous example, no relationship is taken into

a. The **Congress** subcommittee backed away from mandating specific **retaliation against foreign** countries for **unfair** foreign **trade practices**.

b. He urged **Congress** to reject provisions that would mandate U.S. **retaliation against foreign unfair trade practices**.

c. Washington charged France, West Germany, the U.K., Spain and the EC Commission with **unfair practices** on behalf of Airbus.

Figure 2.1: A set of sentences from the Reuters-21578 collection [Reu87].

account between "President John Kennedy" and "President Kennedy".

With maximal frequent sequences, defined in the next section, we allow for an unlimited gap that permits taking language variations into account during the extraction process.

2.1.2 Maximal Frequent Sequences

In this section, we will introduce the concept of a Maximal Frequent Sequence [1]. We will then overview the data mining techniques that aim at the extraction of sequential patterns, and particularly those that permit to extract MFSs.

Definition 1 *A sequence $p = a_1 \cdots a_k$ is a* subsequence *of a sequence q if all the items $a_i, 1 \leq i \leq k$, occur in q and they occur in the same order as in p. If p is a subsequence of q, we also say that p* occurs *in q and that q is a* supersequence *of p.*

For instance, the sequence *"unfair practices"* can be found in all of the three sentences in Figure 2.1.

Definition 2 *A sequence p is* frequent *in a set of fragments S if p is a subsequence of at least σ fragments of S, where σ is a given frequency threshold.*

If we assume that the frequency threshold is 2, we can find the following frequent sequences in our sample set of sentences: *"congress retaliation against foreign unfair trade practices"* and *"unfair practices"* (Fig. 2.1).

Definition 3 *A sequence p is a* maximal frequent (sub)sequence *in a set of fragments S if there does not exist any sequence p' in S such that p is a subsequence of p' and p' is frequent in S.*

In our example, the sequence *"unfair practices"* is not maximal, since it is a subsequence of the frequent sequence *"congress retaliation against foreign unfair trade practices"*. This latter sequence is maximal.

With this simple example, we already get a glimpse of the compact descriptive power of MFSs. Should we be restricted to word pairs, the 7-gram *"congress retaliation against foreign unfair trade practices"* would need to be replaced by $\binom{7}{2} = 21$ bigrams. With MFSs, we can obtain a very compact representation of the regularities of text. The rest of this section will focus on the problem of their efficient extraction in a document collection.

2.1.3 Sequential Pattern Mining in Text

Given a document collection and a minimal frequency threshold, the naïve approach to extract the MFS set is to go through the document collection, collect each frequent word, and use the set of all frequent words to produce candidate word pairs (bigrams) and retain only the frequent ones. The process of forming and counting the frequency of (n+1)-gram candidates from the set of all frequent n-grams can be repeated iteratively as long as frequent (n+1)-grams are found. To obtain the set of all MFSs, it remains to remove every frequent sequence that is a subsequence of another frequent sequence. But this approach is clearly computationally inefficient.

2.1.3.1 Sequential Pattern Mining

Agrawal and Srikant [AS95] introduced the problem of *mining sequential patterns* as an advanced subtask of data mining, where typical data consists of customer transactions, that is, database entries keyed on a *transaction id* and each consisting of a *customer id* associated to the list of items that she bought in this very transaction. The problem of mining sequential patterns is an advanced version of that of the extraction of interesting *item sets*. But in sequential pattern mining, we also aim to exploit the fact that the transaction entries of the databases include a time field that permits to sort the transactions in chronological order and even know the time interval (or distance) that separates them. A motivating example of a sequential pattern, from [AS95], would be that customers typically rent the movie "Star Wars", then "The Empire Strikes Back", and finally "The Return of the Jedi".

Agrawal and Srikant [AS95] presented an improvement of the naïve approach that benefits of an intermediary pruning step to remove all (n+1)-gram candidates that contain at least one non-frequent n-gram. This permits to avoid a number of useless frequency counts. Most approaches are fueled by the same idea of pruning a number of "candidate frequent sequences", to avoid costly frequency counts.

Zaki [Zak01] presented $SPADE$, an advanced technique for the discovery of sequential patterns. Its architecture relies on a vertical database that fastens frequency counts and a lattice-theoretic approach permits to reduce the search space. Unfortunately, the main weakness of $SPADE$ is that it still enumerates all the candidate sequences by forming candidate (n+1)-sequences through the combination of each two n-sequences. DFS_Mine [TG01] was subsequently designed to try to discover n-sequences without enumerating all the frequent sequences of length (n-1). This is done by storing two lists, containing "minimal non-frequent sequences" (because their supersequences are necessarily infrequent) and "maximal frequent sequences" (because their subsequences are necessarily frequent). A significant number of frequency counts can then be avoided. The problem with DFS_Mine is that the candidate (n+1)-sequences are formed by combining an n-sequence with the items of the database. While this may function with spatiotemporal data, the presented application of DFS_Mine, where the number of items is low, this is not reasonable for text, where the number of items (words) can be enormous.

2.1.3.2 Sequential Patterns and Text

The key particularity of text as a sequential data type is the number of items. For instance, the vocabulary of the widely known *Brown corpus* contains 50,406 distinct

words, whereas, e.g., biosequences have a very limited vocabulary: there are only 20 amino acids, and only 4 molecules containing nitrogen in DNA and RNA (A, C, G, and T). Another particularity of text is that the distribution of words is skewed. There is a small number of words that are very frequent, whereas the majority of words are infrequent. The words with moderate frequency are usually considered the most interesting and most informative.

These special characteristics of textual data have a strong influence on the discovery of interesting sequences in text. All the breadth-first, bottom-up approaches are failing quickly for a number of reasons. They permit pruning but require to keep in memory all the subsequences of two distinct lengths. They further generate a large number of candidates whose frequency is slow to count. Depth-first search takes less memory, but the number of items (words) to be intersected with a given sequence is prohibitive.

2.1.3.3 Sequential Pattern Mining in Text: *MineMFS*

MineMFS [1] is a method combining breadth-first and depth-first search that is particularly well-suited for text. It extracts MFSs of any length, i.e., also very long sequences, and it allows an unrestricted gap between words of the sequence. In practice, however, text is usually divided into sentences or paragraphs, which indirectly restricts the length of sequences, as well as the maximal distance between two words of a sequence. The constraints used in the method are minimum and maximum frequency. Hence, words that are less (respectively, more) frequent than a minimum (respectively, maximum) frequency threshold are removed.

Algorithm. As for *DFS_Mine*, an important idea in *MineMFS* is to compute frequent (n+1)-sequences without enumerating all the frequent n-sequences. It relies on a set of "n-gram seeds", initialized with the set of all frequent bigrams. The main idea is to pick an n-gram seed and try to combine it with other grams in a greedy manner, i.e., as soon as the n-gram seed is successfully expanded to a longer frequent sequence, other expansion alternatives are not checked, but only that longer frequent sequence is tentatively expanded again. This expansion procedure is repeated until the longer frequent sequence at hand can only be expanded to infrequent sequences. This sequence is maximal. When all the n-gram seeds have been processed, those that cannot be used to form a new maximal frequent sequence of size more than n are pruned. The remaining ones are joined to produce candidate (n+1)-gram seeds that will be used in a new iteration of the process. This process is repeated until no new maximal frequent sequence can be discovered.

Strengths. A main strength of *MineMFS* versus *DFS_Mine* is the fact that the choice of items that may be inserted to expand an n-gram is restricted to the other non-pruned frequent n-grams. Whereas in *DFS_Mine*, an n-gram is expanded by trying to insert every (or most) frequent word, which is too costly for textual data. Further sophisticated pruning techniques permit restricting the depth-first search, which means only a few alternatives need to be checked to try to expand a sequence, despite the large vocabulary size.

Limitations. Even though the use of minimal and maximal frequency thresholds permits to reduce the burstiness of word distribution, it also causes the miss of a number of truly relevant word associations. For large enough collections, the *MineMFS* process fails to produce results, unless excessive minimal and maximal

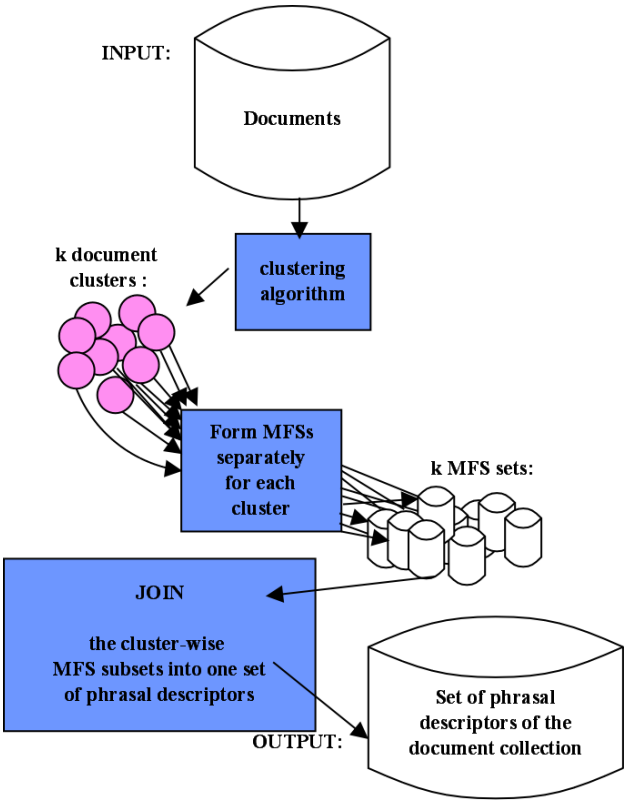

Figure 2.2: The different phases of $MFS\_MineSweep$.

frequencies are decided upon, in which case the set of MFSs produced is small and contains mostly non-interesting descriptors. One reason may be the pruning step, which runs through the set of n-grams and compares each two of them that may form an $(n+1)$-gram, by checking if a new item can be added between every two adjacent words. The number of possible positions of insertion shall be problematic.

2.1.4 A Divide and Conquer Approach: $MFS\_MineSweep$

We have seen that $MineMFS$ fails to extract the MFS set of a sufficiently large document collection, we proposed $MFS\_MineSweep$ [16, 36], a technique to decompose a collection of documents into several disjointed subcollections, small enough so that the MFS set of each subcollection can be extracted efficiently. Joining all the sets of MFSs, we obtain an approximate of the maximal frequent sequence set for the full collection. $MFS\_MineSweep$ permits extracting more and sharper descriptors from document collections of virtually any size. Its main drawback is the loss of the maximality property, producing a less compact set of content descriptors.

2.1.4.1 Description and Motivation

Our approach relies on the idea to partition the document collection into a set of homogeneous subcollections. The initial motivation to do this is that $MineMFS$ does not produce any result at all for sufficiently large document collections. Figure 2.2 describes the steps of $MFS\_MineSweep$. In the first phase, we apply $MineMFS$

- d_1: Mary had a little lamb whose fleece was white as snow.
- d_2: A radio station called Sputnik broadcasts Russian programs in Saint-Petersburg and Helsinki. It was named after the first satellite ever launched.
- d_3: History changed on October 4, 1957, when the Soviet Union successfully launched Sputnik I. The world's first artificial satellite was about the size of a basketball, weighed only 183 pounds, and revolved around the Earth in about 98 minutes.
- d_4: Everywhere that Mary went, her lamb was sure to go.

Figure 2.3: A collection of four documents.

on a number of disjoint subcollections, so as to obtain an MFS set corresponding to each subcollection. The second step is to gather the MFS sets of each subcollection to form a set of content descriptors for the whole collection. This gathering operation mainly consists in appending the sets of MFSs, as there is no clear way to join a sequence (maximal frequent in a subcollection) to its subsequence (maximal frequent in another). Only identical sequences can be merged. Thus, the maximality property is lost, and therefore, the content description of our pre-partitioning technique is always less or equally compact to that of the MFSs of the whole collection.

We ran a number of experiments to verify the validity of the $MFS\_MineSweep$ algorithm.

The main motivation for developing $MFS\_MineSweep$ was to efficiently obtain a more detailed description of the document collection, as we could use looser frequency thresholds. This is easily understood by thinking of an extreme case; if a collection of $|D|$ documents is split into $|D|$ subcollections of size 1 and the minimal frequency is 1, we can obtain the corresponding sets of MFS instantly: each MFS set contains only one sequence of frequency 1, the unique document in the corresponding subcollection. No information is lost, but the content description is probably too large.

We also conjectured that more consistent subcollections permit to obtain better descriptors. The main reason of this train of thought relies on the fact that a collection made of similar documents will contain more interesting MFSs than a collection made of dissimilar documents. Again, thinking of extreme cases makes this point easier to see, as a collection where no two documents have a word in common will not contain any frequent sequences, except for the documents themselves (if the frequency threshold is 1).

For example, let us assume that we want to partition the collection of four documents presented in Figure 2.3 into 2 subcollections of 2 documents each, and use a minimal frequency of 2 for extracting MFSs from the subcollections. Only by clustering together the similar documents (d_1, d_4) and (d_2, d_3), will we obtain sequences of words, that is, *phrasal descriptors*. Those descriptors are: "Mary lamb was" for d_1 and d_4, and "Sputnik first satellite launched" for d_2 and d_3. Any other way to partition the collection produces an empty *phrasal description*.

2.1.4.2 Definition of Metrics to Characterize Sequential Descriptions

To experimentally confirm or disprove these hypotheses, we needed measures to compare different sets of phrasal descriptors. Ideal metrics upon which to compare sets of descriptors need to be able to evaluate two things: 1) the size of the phrasal text representation, and 2) the amount (and density) of information it contains.

In general, the problem of comparing two sets is not an easy one. A large quantity of work in the domains of document clustering and textual classification has proposed measures to compare different ways to partition document sets [Seb02]. Unfortunately, we could not exploit this work to solve our problem, because such techniques rely on the comparison of a given clustering (or classification) to a gold standard. In the general case of textual representation, without aiming at a specific application, there is no clear way to define a gold standard of the phrasal description of a document collection.

Fortunately, the problem we were facing is a sub-problem of the above. The sets we needed to compare were indeed similar in nature. For example, a major difficulty in comparing general sequences would be the comparison of long grams to their subgrams. However, in the specific case where all the descriptors are MFS (either of the whole collection or of one of its subcollections), we can simplify the problem by normalizing each descriptor to a set of all its subpairs. This is because the unlimited distance allowed between any two words of an MFS ensures that the assertion "$ABCD$ is an MFS" implies "AB, AC, AD, BC, BD, and CD are frequent bigrams".

We can thus transform each set of phrasal descriptors into a set of comparable items, the frequent bigrams it contains. Let R_D be the phrasal description of a document collection D, and R_d be the corresponding set of phrases describing a document $d \in D$. We can write the corresponding set of word pairs as $bigrams(R_d)$. For $b \in bigrams(R_d)$, we also define df_b as the document frequency of the bigram b. Finally, we define the random variable X over the set $bigrams(R_d)$. For all $b \in bigrams(R_d)$:

$$p(X = b) = \frac{df_b}{\sum_{y \in \{ \bigcup_{d \in D} bigrams(R_d) \}} df_y},$$

where $\sum_{y \in \{ \bigcup_{d \in D} bigrams(R_d) \}} df_y$ is the total number of bigram occurrences resulting from the phrasal description R_D. It can be thought of as the sample size.

Size of the representation of a document collection. The phrasal representation of a document collection can be seen as a set of associations between descriptive n-grams and documents. We define $|R_D|$ as the size of the phrasal representation R_D in a very intuitive way:

$$|R_D| = \sum_{d \in D} |R_d|.$$

Hence, $|R_D|$ is the number of document-phrase associations in the collection representation R_D.

Implied quantity of frequent bigrams in the representation. Several phrases may contain identical bigrams that represent the same document. To count the num-

ber of implied document-bigram associations permits to ignore redundant information stemming from the long descriptors. We shall therefore measure the quantity of information in the description with the number of document-bigram associations that correspond to the description R_D. This value is $bigram\_size(R_D)$, defined as follows:

$$bigram\_size(R_D) = \sum_{d \in D} |bigrams(R_d)|.$$

Hence, $bigram\_size(R_D)$ is the number of document-bigram associations stemming from the collection representation R_D.

Density of the description. To measure whether the description is loose or dense, we can use the two preceding metrics in a very simple way. By computing the ratio between the number of document-bigram associations in a document representation and its size, we obtain a relative measure of the number of document-bigram associations that can be avoided with longer n-grams:

$$Density(R_D) = \frac{bigram\_size(R_D)}{|R_D|}.$$

For example, a density value of 1.1 means that the bigram representation of R_D contains 10% more associations than the equivalent representation R_D. The higher $Density(R_D)$, the more storage space we save by using R_D instead of frequent pairs only.

2.1.4.3 Findings about $MFS\_MineSweep$

It is important to observe that the extraction of the set of MFSs is an independent process for each distinct subcollection. A profitable alternative is to run the extraction of the MFS sets in parallel, on distinct computers. The total running time is then the time of the slowest MFS set extraction, plus the time for splitting the original document collection. We ran experiments on a set of desktops with a 2.80 Ghz processor and 1024Mb of RAM and experimented with the 16Mb Reuters-21578 newswire collection [Reu87], which originally contains about 19,000 non-empty documents.

To place both techniques on equal grounds, we found a frequency range for every subcollection individually, such that the corresponding MFS extraction time is always between 4 and 5 minutes. This was achieved with a fairly simple heuristic, interrupting the process and decreasing the frequency range when the extraction was too slow, and increasing the frequency range after too fast an extraction. We then compare the resulting sizes, amounts and densities of information.

MFS_MineSweep outperforms MineMFS. Our first observation was that both the number of descriptors and the number of equivalent bigrams are always much higher for $MFS\_MineSweep$ than for $MineMFS$. These numbers increase with the number of partitions.

The description is less compact. Consequently, the density of the phrasal representations is decreasing with the number of subcollections. What we did not expect is that the density ratio goes down to values below 1, meaning that the number of equivalent bigrams is less than the number of phrasal descriptors. This steep density decrease expresses more than the loss of the maximality property. A lower density means that the number of descriptors is growing faster than the number of bigrams. When we split the collection into more disjoint subcollections, this means that more and more of the new descriptors we find are only new combinations of bigrams that we already found when we split the collection in less partitions. This sharp decrease in density is in fact an indication that the discriminative power of the phrasal description is peaking, and that further augmentations of the number of partitions will be comparatively less and less worthwhile.

The more homogeneous the subcollections, the better the descriptors. To determine whether clustering was beneficial, we compared the size, amount and density of information obtained when splitting the collection into random and homogeneous subcollections.

Homogeneity provides better discrimination. We found out that when the number of partitions rises, the density of the description resulting from homogeneous subcollections decreases slowly, whereas the steep is much sharper for random partitions. The fact that the description densities resulting from homogeneous collections remain nearly stable shows that there is room to improve the discriminative power of phrasal descriptions if we partition the document collection in even more clusters. The reason is simple. The descriptors extracted from random subcollections are ones that are present all over the collection. Splitting the collection into more subsets permits finding more of those frequent n-grams, formed by the same frequent words, but we reach a point where we only find combinations of the same frequent words originating from different subcollections. On the other hand, homogeneous subcollections permit gathering similar documents together, excluding non-similar documents. Hence, the frequency range can be adapted to extract the specifics of each subcollection. In the homogeneous case, increasing the number of subcollections permits embracing more specificities of the document collections, whereas in the random case, it only permits catching more descriptors of the same kind.

Clustering is safer. As opposed to random partitioning, clustering provides *guarantees*. It is more reliable, because it ensures result. The strength of random partitioning is it gives good results and permits MFS extraction in predictable times. But these facts are only true *on average*. The problem if we use random partitioning is that we should, in fact, run several iterations to protect ourselves from an "unlucky" draw. We mentioned earlier that running several random iterations increases the exposure to factors of difficult extraction. Another issue with averaging numerous iterations is practical. Assume document d was represented 3 times by $gram_A$, and 1 time by $gram_B$ and $gram_C$, what should be the average document description of d? Because the extraction of MFS sets from homogeneous subcollections is unique and needs to be done only once, it is generally less costly.

2.1.5 Conclusion and Perspectives

We have introduced the concept of a maximal frequent sequence, a compact approach to document description. We presented a number of techniques that permit to extract MFSs from sequential data, before covering their weaknesses, when applied to textual data. An efficient solution was introduced with $MineMFS$, although it still fails to produce descriptors efficiently for too large document collections.

We consequently presented $MFS\_MineSweep$, a technique to obtain a better description efficiently, by running $MineMFS$ on homogeneous partitions of the document collection, and joining the results. We introduced measures of quantity, size and density of information to compare results obtained by lone use of $MineMFS$ to those obtained by its use within the $MFS\_MineSweep$ framework. This confirmed that $MFS\_MineSweep$ permits the extraction of more exhaustive descriptions, faster.

Possible extensions. To improve the share of true multiword units within the extracted MFSs, one possibility is to attach POS tags prior to the extraction process, and use those tags for pattern-based filtering, in a similar fashion as what was proposed in Smadja's Xtract [Sma93].

Another way to improve the linguistic meaningfulness of the extracted MFSs, without compromising the language independence of the process (by, e.g., requiring a POS tagger), is to exploit the structural markup of documents. We experimented in document retrieval and classification with such structure-fed phrasal descriptors. This work is presented in the present manuscript's chapter focused on the use of extracted knowledge, specifically in Sections 3.3 and 3.4.

Direct evaluation of sequences. A purely statistical technique to rank (therefore to filter) extracted sequences is presented in the following section. The *general* evaluation of individual descriptors is a difficult problem. In numerous real-life applications, it is crucial to be able to rank or weight phrasal descriptors. Basic approaches, such as using the rough length or frequency of the word sequences appear insufficient. In the following section, we will present an advanced technique for calculating the probability of occurrence, document frequency, and general-purpose interestingness of discontiguous sequences of any length.

2.2 Filtering Discontinued Sequences

The probability of occurrence of words and phrases is a crucial matter in all domains of information retrieval. All language models rely on such probabilities. However, while the probability of a word is frequently based on counting its total number of occurrences in a document collection (its *collection frequency*), calculating the probability of a phrase is far more complicated. Counting the number of occurrences of a multiword unit is often intractable, unless restrictions are adopted, such as setting a maximal unit size, requiring word adjacency or setting a maximal distance between two words.

This section presents an efficient technique for calculating the probability of occurrence of a *discontinued* sequence of words, i.e., the probability that those words

occur, and that they occur in a given order, regardless of which and how many other words may occur between them. The procedure we introduce for words and documents may be generalized to any type of sequential data, e.g., item sequences and transactions. Our method relies on the formalization into a particular Markov chain model, whose specificities are combined with techniques of probability and linear algebra to offer competitive computational complexity. This work is further extended to permit the efficient calculation of the *expected document frequency* of a sequence. An application is a fast, automatic, and direct method to evaluate the interestingness of word sequences, by comparing their expected and observed frequencies through straightforward statistical testing.

This technique permits to efficiently calculate the exact probability (respectively, the expected document frequency) of a given sequence of n words to occur in a document of size l, (respectively, in a document collection D) with an unlimited number of other words eventually occurring between them. We assume that words occur independently, i.e., the probability of occurrence of a word in a given position does not depend on its context.

An application of this result is a *fast and automatic technique to directly evaluate the interestingness* of word sequences. Phrase extraction techniques often output a number of uninteresting sequences and it is desirable to have means to sort them by their level of interestingness. One main advantage of a ranked list over a set of phrasal descriptors is that it permits to the end-user to save time by reading through the most important findings first. This is especially important in real-life applications, where time is a limited resource. To rank a list of phrasal descriptors is not trivial, especially when it comes to comparing phrases of different lengths.

By exploiting statistical techniques, of *hypothesis testing*, our method provides the ability to do exactly that. The main idea is to account for the fact that word sequences are bound to happen by chance, and to compare how often a given word sequence *should* occur to how often it truly does. That is, the more the actual number of occurrences of a phrase is higher than its expected frequency, the stronger the lexical cohesion of that phrase. This evaluation technique is entirely language-independent, as well as domain- and application-independent. It permits to efficiently rank a set of candidate multiword units, based on statistical evidence, without requiring manual assessment of a human expert.

In the following sections, we will introduce the problem, present the naïve technique to calculate the probability of an n-words sequence to occur in a document, before summarizing our technique, including a complexity analysis that shows how it outperforms naïve approaches. We will show how to generalize the probability of occurrence of an n-words sequence into its expected document frequency in a document collection, with a very reasonable computational complexity.

2.2.1 Probability of Discontinued Occurrence of an n-Words Sequence

It should be clear to the reader that the method is only sketched here, and the proofs are essentially discarded. For reference and full details, I recommend the article published in 2006 in the journal "*Traitement Automatique des Langues* (TAL)" and entitled "Probability and Expected Document Frequency of Discontinued Word

Sequences, an Efficient Method for their Exact Computation" [9].

2.2.1.1 Problem Definition

Let $A_1, A_2, \ldots, A_n$ be n words, and d a document of length l (i.e., d contains l word occurrences). Each word A_i is assumed to occur independently with probability p_{A_i}.

Problem. In d, we want to calculate the probability $P(A_1 \to A_2 \to \cdots \to A_n, l)$ of the words A_1, A_2, ..., A_n to occur at least once in this order, an unlimited number of interruptions of any size being permitted between each A_i and A_{i+1}, $1 \leq i \leq (n+1)$.

More definitions. Let D be the document collection, and W the set of all distinct words occurring in D. The calculation of the probability p_w of occurrence of a word w is a vast problem. In this work, we assume these probabilities to be given. In our experiments and along the examples, we use the term frequency of w in the whole document collection, divided by the total number of word occurrences in the collection. One reason to choose this approach is that the set of all word probabilities $\{p_w \mid \forall w \in W\}$ is then a (finite) probability space. Indeed, we have:

$$\sum_{w \in W} p_w = 1, \text{ and } p_w \geq 0, \forall w \in W.$$

For convenience, we will also simplify the notation of p_{A_i} to p_i, and define $q_i = 1 - p_i$, the probability of non-occurrence of the word A_i.

A running example. Let there be a hypothetic document collection containing only three different words A, B, and C, each occurring with equal frequency. We want to find the probability that the bigram $A \to B$ occurs in a document of length 3.

For such a simple example, we can afford an exhaustive enumeration. There exist $3^3 = 27$ distinct documents of size 3, each occurring with equal probability $\frac{1}{27}$. These documents are:

$$\{AAA, \boxed{\text{AAB}}, AAC, \boxed{\text{ABA}}, \boxed{\text{ABB}}, \boxed{\text{ABC}}, ACA, \boxed{\text{ACB}}, ACC,$$
$$BAA, \boxed{\text{BAB}}, BAC, BBA, BBB, BBC, BCA, BCB, BCC,$$
$$CAA, \boxed{\text{CAB}}, CAC, CBA, CBB, CBC, CCA, CCB, CCC\}$$

The 7 framed documents contain the n-gram AB. Thus, we have $p(A \to B, 3) = \frac{7}{27}$.

2.2.1.2 Naïve Computation

The main naïve approach relies on a way to categorize the different sets of documents of size l in which the n-gram $A_1 \to \cdots \to A_n$ occurs, with the property that all the sets are disjoint and that no case of occurrence of the n-gram is forgotten. This ensures that we can calculate $p(A_1 \to \cdots \to A_n, l)$ by summing up the probabilities of each set of documents where $A_1 \to \cdots \to A_n$ occurs.

A Disjoint Categorization of Successful Documents. We can indeed split the *successful* documents (those in which the n-gram occurs) of size l, depending on the position from which a successful outcome is guaranteed. For example, and for $l \geq n$, the documents of size l for which success is guaranteed from position k onwards can be represented by the set of documents E_k, for $0 \leq k \leq l-n$, occurring with probability $p(E_k)$:

$$p(E_k) = \prod_{i=1}^{n} p_i \sum_{i_n=0}^{k} \cdots \sum_{i_2=0}^{k-(i_n+\cdots+i_3)} q_1^{k-\sum_{j=2}^{n} i_j} q_2^{i_2} \ldots q_n^{i_n}.$$

Probability of the n-words sequences. Because in any document, the presence of the n-gram is ensured from only one position onwards, it is clear that the sets E_k, for $0 \leq k \leq (l-n)$, are all disjoint. It is also evident that in any document of size l containing the n-gram, its occurrence will be ensured between the n-th and l-th position. Therefore the sets E_k are mutually exclusive, for $0 \leq k \leq (l-n)$, and their union contains all the documents of size l where $A_1 \to \cdots \to A_n$ occurs. Consequently, the formula of the probability of occurrence of a discontiguous sequence of length n in a document of length l is:

$$p(A_1 \to \cdots \to A_n, l) = \prod_{i=1}^{n} p_i \sum_{i_n=0}^{l-n} \cdots \sum_{i_1=0}^{l-n-(i_n+\cdots+i_2)} q_1^{i_1} q_2^{i_2} \ldots q_n^{i_n}. \tag{2.1}$$

Running Example. For better comprehension, let us return to the running example:

$$\begin{aligned} p(A \to B, 3) &= p_a p_b \sum_{i_b=0}^{1} \sum_{i_a=0}^{1-i_b} q_a^{i_a} q_b^{i_b} \\ &= p_a p_b \left(1 + q_a + q_b\right) \\ &= \frac{1}{3} \times \frac{1}{3} \times \left(1 + \frac{2}{3} + \frac{2}{3}\right) \\ &= \frac{7}{27}. \end{aligned}$$

We indeed find the exact result.

Computational Complexity. Unfortunately, the order of complexity of the direct computation of Formula [2.1] is $O(ln^{l-n})$. Consequently, this formula is hardly usable at all, except for short documents and length-restricted n-grams.

2.2.2 Efficient Computation through a Markov Chain Formalization

While the previous method was our initial contribution, we later found a much more efficient technique by means of a slightly different formalization.Let us consider the problem as a sequence of l trials whose outcomes are $X_1, X_2, \ldots, X_l$. Let each of these outcomes belong to the set $\{0, 1, \ldots, n\}$, where the outcome i signifies that the i-gram $A_1 \to A_2 \to \cdots \to A_i$ has already occurred. This sequence of trials verifies the following two properties:

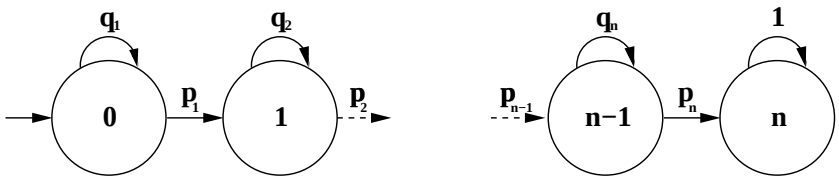

Figure 2.4: The state-transition diagram of the Markov Chain M.

(i) All the outcomes $X_1, X_2, \ldots, X_l$ belong to a finite set of outcomes $\{0, 1, \ldots, n\}$ called the *state space* of the system. If i is the outcome of the m-th trial ($X_m = i$), then we say that the system is in state i at the m-th step. In other words, the i-gram $A_1 \to A_2 \to \cdots \to A_i$ has been observed after the m-th word of the document.

(ii) The second property is called the *Markov property*: the outcome of each trial depends at most upon the outcome of the immediately preceding trial, and not upon any other previous outcome. In other words, *the future is independent of the past, given the present.* This is verified indeed; if we know that we have seen $A_1 \to A_2 \to \cdots \to A_i$, we only need the probability of A_{i+1} to determine the probability that we will see more of the desired n-gram during the next trial.

These two properties are sufficient to call the defined stochastic process a (finite) *Markov chain*. The problem can thus be represented by an $(n+1)$-states Markov chain M (see Figure 2.4). The state space of the system is $\{0, 1, \ldots, n\}$ where each state, numbered from 0 to n tells how much of the n-gram has already been observed. Presence in state i means that the sequence $A_1 \to A_2 \to \cdots \to A_i$ has been observed. Therefore, $A_{i+1} \to \cdots \to A_n$ remains to be seen, and the following expected word is A_{i+1}. It will be the next word with probability p_{i+1}, in which case a state transition will occur from i to $(i+1)$. A_{i+1} will not be the following word with probability q_{i+1}, in which case we will remain in state i. Whenever we reach state n, we can denote the experience a success: the whole n-gram has been observed. The only outgoing transition from state n leads to itself with associated probability 1 (such a state is said to be *absorbing*).

Stochastic Transition Matrix (in general). Another way to represent this Markov chain is to write its transition matrix. For a general finite Markov chain, let $p_{i,j}$ denote the transition probability from state i to state j for $1 \leq i, j \leq n$. The (one-step) stochastic transition matrix is:

$$P = \begin{pmatrix} p_{1,1} & p_{1,2} & \cdots & p_{1,n} \\ p_{2,1} & p_{2,2} & \cdots & p_{2,n} \\ & & \cdots & \\ p_{n,1} & p_{n,2} & \cdots & p_{n,n} \end{pmatrix}.$$

Theorem 2.2.1 *[Fel68] Let P be the transition matrix of a Markov chain process. Then the m-step transition matrix is equal to the m-th power of P. Furthermore, the entry $p_{i,j}(m)$ in P^m is the probability of stepping from state i to state j in exactly m transitions.*

Our stochastic transition matrix of interest. For the Markov chain M defined above, the corresponding stochastic transition matrix is the following $(n+1)\times(n+1)$ square matrix:

$$M = \begin{array}{c} \text{states} \\ 0 \\ 1 \\ \\ \\ n \end{array} \begin{array}{c} \begin{array}{ccccc} 0 & 1 & \dots & n-1 & n \end{array} \\ \begin{pmatrix} q_1 & p_1 & \dots & \dots & 0 \\ 0 & q_2 & \ddots & & \vdots \\ \vdots & \ddots & \ddots & \ddots & \vdots \\ \vdots & & \ddots & q_n & p_n \\ 0 & \dots & \dots & 0 & 1 \end{pmatrix} \end{array}.$$

Therefore, the probability of the n-gram $A_1 \to A_2 \to \cdots \to A_n$ to occur in a document of size l is the probability of stepping from state 0 to state n in exactly l transitions. Following Theorem 2.2.1, this value resides at the intersection of the first row and the last column of the matrix M^l:

$$M^l = \begin{pmatrix} m_{1,1}(l) & m_{1,2}(l) & \dots & \boxed{m_{1,n+1}(l)} \\ m_{2,1}(l) & m_{2,2}(l) & \dots & m_{2,n+1}(l) \\ & & \dots & \\ m_{n+1,1}(l) & m_{n+1,2}(l) & \dots & m_{n+1,n+1}(l) \end{pmatrix}.$$

Thus, the result we are aiming at can simply be obtained by calculating M^l, and looking at the value in the upper-right corner. Doing so results in a time complexity of $O(ln^3)$.

Alternatively, this computation can be achieved through more time-efficient algorithms for matrix multiplication. The lowest exponent currently known is $O(n^{2.376})$ [CW87]. The strong drawback of such techniques is, however, the presence of a constant so large that it removes the benefits of the lower exponent for all practical sizes of matrices [HJ94]. For our purpose, the use of such an algorithm is typically more costly than to use the naive $O(n^3)$ matrix multiplication.

Exploiting specificities of matrix M. Linear algebra techniques, and a careful exploitation of the specificities of the stochastic matrix M will, however, permit to perform a few transformations that will drastically reduce the computational complexity of raising the matrix M to the power of l.

We relied on the Jordan Normal Form [ND77] of the matrix M and proved that in the Jordan Normal Form of M, there exists one and only distinct Jordan block for every distinct q_i (and that its size equals the number of occurrences of q_i in the main diagonal of M), plus a block of size 1 for the eigenvalue 1. In addition, because a Jordan block is defined as the sum of a diagonal matrix and a nilpotent matrix, raising it to the power of l is fairly simple (please refer once more to the aforementioned article [9] for full details).

As a consequence, we could write M^l as

$$M^l = SJ^lS^{-1} = S \begin{pmatrix} \boxed{J_{e_1}^l} & & 0 \\ & \ddots & \\ 0 & & \boxed{J_{e_q}^l} \end{pmatrix} S^{-1},$$

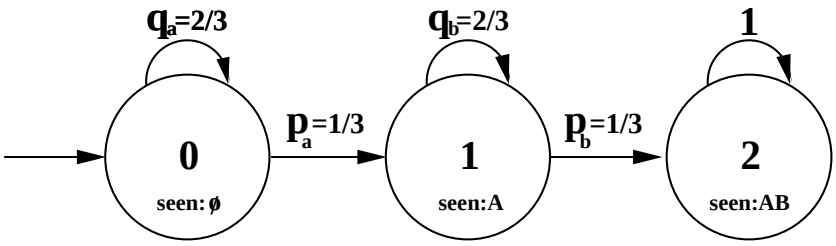

Figure 2.5: The state-transition diagram of the Markov Chain corresponding to our running example.

where the $J^l_{e_i}$ are Jordan blocks, and S and S^{-1} are obtained through the Jordan Normal Form theorem [ND77]. The $J^l_{e_i}$ can further be written as follows [9]:

$$J^l_{e_i} = \begin{pmatrix} \binom{l}{0} \cdot e_i^l & \dots & \binom{l}{k} \cdot e_i^{l-k} & \dots & \binom{l}{n_i-1} \cdot e_i^{l-n_i+1} \\ & \ddots & & \ddots & \vdots \\ & & \binom{l}{0} \cdot e_i^l & & \binom{l}{k} \cdot e_i^{l-k} \\ & & & \ddots & \vdots \\ 0 & & & & \binom{l}{0} \cdot e_i^l \end{pmatrix}.$$

To calculate (the upper-right value of) M^l therefore only requires to calculate and multiply (the first row of) S, J^l and (the last column of) S^{-1}. Before discussing the complexity of this approach, we will return to the running example presented in Subsection 2.2.1.1.

Running Example. The state-transition diagram of the Markov Chain corresponding to the bigram $A \to B$ has only three states (see Figure 2.5). The corresponding transition matrix is:

$$M_{re} = \begin{pmatrix} \frac{2}{3} & \frac{1}{3} & 0 \\ 0 & \frac{2}{3} & \frac{1}{3} \\ 0 & 0 & 1 \end{pmatrix}.$$

Following the Jordan normal form theorem, there exists an invertible matrix S_{re} such that

$$J_{re} = S_{re}^{-1} M_{re} S_{re} = \begin{pmatrix} \boxed{J_{\frac{2}{3}}} & 0 \\ 0 & \boxed{J_1} \end{pmatrix},$$

where J_1 is a block of size 1, and $J_{\frac{2}{3}}$ a block of size 2 since $q_a = q_b = \frac{2}{3}$. We can actually write J_{re} as:

$$J_{re} = \begin{pmatrix} \frac{2}{3} & 1 & 0 \\ 0 & \frac{2}{3} & 0 \\ 0 & 0 & 1 \end{pmatrix}.$$

Since we seek the probability of the bigram $A \to B$ in a document of size 3, we need to calculate J^3_{re}:

$$J^3_{re} = \begin{pmatrix} \binom{3}{0}(\frac{2}{3})^3 & \binom{3}{1}(\frac{2}{3})^2 & 0 \\ 0 & \binom{3}{0}(\frac{2}{3})^3 & 0 \\ 0 & 0 & 1 \end{pmatrix} = \begin{pmatrix} \frac{8}{27} & \frac{4}{3} & 0 \\ 0 & \frac{8}{27} & 0 \\ 0 & 0 & 1 \end{pmatrix}.$$

Further details as to the practical computation of S_{re} and the last column of its inverse S_{re}^{-1} are available in the following section on computational complexity and in [9]. For now, let us simply assume they were calculated, and thus:

$$P(A \to B, 3) = \overbrace{(1\ 0\ 1)}^{\text{first row of } S} \begin{pmatrix} \frac{8}{27} & \frac{4}{3} & 0 \\ 0 & \frac{8}{27} & 0 \\ 0 & 0 & 1 \end{pmatrix} \overbrace{\begin{pmatrix} -1 \\ -\frac{1}{3} \\ 1 \end{pmatrix}}^{\text{last column of } S^{-1}} = \frac{7}{27}.$$

Our technique indeed obtains the right result. But how efficiently is it obtained? The purpose of the following section is to answer this question.

2.2.2.1 Algorithmic Complexity

The process of calculating the probability of occurrence of an n-gram in a document of size l consists of the following phases: calculating J^l, computing the transformation matrix S and (the last column) of its inverse S^{-1}, and multiplying the first row of S by M, and the result by the last column of S^{-1}.

In my doctoral work, and in the journal article article aforementioned [9, 36], it was demonstrated that the complexity of the computation of J^l is $O(lq)$, and that the calculation of the transformation matrix S can be achieved in $O(n^2)$.

While the general inversion of a matrix is done in $O(n^3)$ through Gaussian elimination, we could also prove that S, in this particular case, is always a column permutation of an upper-triangular matrix. This implied that the whole process of computing the last column of the matrix S^{-1} is $O(n^2)$.

Conclusion. As we have seen, directly raising M to the power of l is $O(ln^3)$, while the computation of the exact mathematical Formula [2.1] is only achieved in $O(ln^{l-n})$. However, following our technique, **an upper bound** of the complexity for computing **the probability of occurrence of an n-gram in a document of size l is** $O(ln)$.

2.2.2.2 The Expected Frequency of an n-Words Sequence

After we had defined a formula to calculate the probability of occurrence of an n-gram in a document of size l, we could use it to calculate the expected document frequency of the n-gram in the whole document collection D. Assuming the documents are mutually independent, the expected frequency in the document collection is the sum of the probabilities of occurrence in each document:

$$Exp\_df(A_1 \to \cdots \to A_n, D) = \sum_{d \in D} p(A_1 \to \cdots \to A_n, |d|),$$

where $|d|$ stands for the number of word occurrences in the document d.

Naïve Computational Complexity. We can compute the probability of an n-gram to occur in a document d in $O(|d|n)$. A separate computation and summation of the values for each document can thus be computed in $O(\sum_{d \in D} |d|n)$.

Better Computational Complexity. However, we achieved better complexity by summarizing everything we needed to calculate and organizing the computation in a sensible way. Let $L = \max_{d \in D} |d|$ be the size of the longest document in the collection and $|D|$, the number of documents in D. We first need to raise the Jordan matrix J to the power of every distinct document length, and then to multiply the (at worst) $|D|$ distinct matrices by the first row of S and the resulting vectors by the last column of its inverse S^{-1}.

The matrix S and the last column of S^{-1} need to be computed only once, and as we have seen previously, this is achieved in $O(n^2)$, whereas the $|D|$ multiplications by the first row of S are done in $O(|D|nq)$. It now remains to find the computational complexity of the various powers of J.

We must first raise each eigenvalue e_i to the power of L, which is an $O(Lq)$ process. For each document $d \in D$, we obtain all the terms of $J^{|d|}$ by $(n+1)$ multiplications of powers of eigenvalues by a set of combinatorial coefficients computed in $O(\max_{block})$. The total number of such multiplications is thus $O(|D|n)$, an upper bound for the computation of all combinatorial coefficients. The worst case time complexity for computing the set $\{\ J^{|d|} \mid d \in D\}$, is then $\max\{O(|D|n), O(Lq)\}$.

Finally, **the computational complexity for calculating the expected frequency of an n-gram in a document collection D is** $\max\{O(|D|nq), O(Lq)\}$, where q is the number of words in the n-gram having a distinct probability of occurrence, and L is the size of the longest document in the collection. The improvement is considerable, compared to the computational complexities of the more naive techniques, in $O(\sum_{d \in D} |d| n^{l-n})$ and $O(\sum_{d \in D} |d| n^3)$.

2.2.3 Direct Evaluation of Lexical Cohesive Relations

In this section, we will introduce an application of the expected document frequency that fills a gap in information retrieval. We propose a direct technique, language- and domain-independent, to rank a set of phrasal descriptors by their interestingness, *regardless of their intended use.*

The evaluation of lexical cohesion is a difficult problem. Attempts at direct evaluation are rare, simply due to the subjectivity of any human assessment, and to the wide acceptance that we first need to know what we want to do with a lexical unit before being able to decide whether or not it is relevant for that purpose. A common application of research in lexical cohesion is lexicography, where the evaluation is carried out by human experts who simply look at phrases to assess them as good or bad. This process permits scoring the extraction process with highly subjective measures of precision and recall. However, a linguist interested in the different forms and uses of the auxiliary "to be" will have a different view of what is an interesting phrase than a lexicographer. What a human expert judges as uninteresting may be highly relevant to another.

Hence, most evaluation has been indirect, through question-answering, topic segmentation, text summarization, and passage or document retrieval [Vec05]. To pick the last case, such an evaluation consists in trying to figure out which are the phrases that permit to improve the relevance of the list of documents returned. A weakness of indirect evaluation is that it hardly shows whether an improvement is due to the quality of the phrases, or to the quality of the technique used to exploit

them. Moreover, text retrieval collections often have a relatively small number of queries, which means that only a small proportion of the phrasal terms will be used at all. This is a strong argument against the use of text retrieval as an indirect way to evaluate the quality of a phrasal index, initially pointed out by Fox [Fox83].

There is a need to fill the lack of a general purpose direct evaluation technique, one where no subjectivity or knowledge of the domain of application will interfere. Our technique permits exactly that, and we will now explain how.

2.2.3.1 Hypothesis testing

A general approach to estimate the interestingness of a set of events is to measure their statistical significance. In other words, by evaluating the validity of the assumption that an event occurs only by chance (the *null hypothesis*), we can decide whether the occurrence of that event is interesting or not. If a frequent occurrence of a multiword unit was to be expected, it is less interesting than if it comes as a surprise.

To estimate the quality of the assumption that an n-gram occurs by chance, we need to compare its (by chance) expected frequency and its observed frequency. There exists a number of tests, extensively described in statistics textbooks, even so in the specific context of natural language processing [MS99]. We chose to base our experiments on the *t-test*:

$$t = \frac{Obs\_df(A_1 \rightarrow \cdots \rightarrow A_n, D) - Exp\_df(A_1 \rightarrow \cdots \rightarrow A_n, D)}{\sqrt{|D| Obs\_DF(A_1 \rightarrow \cdots \rightarrow A_n)}}$$

2.2.3.2 Experiments

We ran experiments with Maximal Frequent Sequences extracted from the publicly available Reuters-21578 newswire collection [Reu87], which originally contains about 19,000 non-empty documents. We split the data into 106,325 sentences. The average size of a sentence is 26 word occurrences, while the longest sentence contains 260.

Using $MineMFS$ with a minimum frequency threshold of 10, we obtained 4,855 MFSs, distributed in 4,038 2-grams, 604 3-grams, 141 4-grams, and so on. The longest sequences had 10 words.

The expected document frequency and the t-test of all the MFSs were computed in 31.425 seconds on a laptop with a 1.40 Ghz processor and 512Mb of RAM.

Results. Table 2.1 shows the overall best-ranked MFSs. The number in parenthesis after each word is its frequency. With Table 2.2, we can compare the best-ranked bigrams of frequency 10 to their worst-ranked counterparts (which are also the worst-ranked n-grams overall), noticing a difference in quality that the observed frequency alone does not reveal.

It is important to note that **our technique permits to rank long n-grams amongst shorter ones**. For example, the best-ranked n-gram of a size higher than 2 lies in the 10^{th} position: *"chancellor exchequer nigel lawson"* with t-test value 0.02315, observed frequency 57, and expected frequency $0.2052e-07$.

t-test	n-gram	expected	observed
.0311	los(127) angeles(109)	.0809	103
.0282	kiichi(88) miyazawa(184)	.0946	85
.0274	kidder(91) peabody(94)	.0500	80
.0267	morgan(382) guaranty(93)	.2073	76
.0249	latin(246) america(458)	.6567	67
.0243	orders(516) orders(516)	1.550	66
.0243	leveraged(85) buyout(145)	.0720	63
.0240	excludes(350) extraordinary(392)	.7995	63
.0239	crop(535) crop(535)	1.666	64
.0232	chancellor(120) exchequer(100) nigel(72) lawson(227)	2.052e-8	57

Table 2.1: Overall 10 best-ranked MFSs

t-test	n-gram	expected	observed
9.6973-3	het(11) comite(10)	.6430-3	10
9.6972-3	piper(14) jaffray(10)	.8184-3	10
9.6969-3	wildlife(18) refuge(10)	.0522-3	10
9.6968-3	tate(14) lyle(14)	.1458-3	10
9.6968-3	g.d(10) searle(20)	.1691-3	10
8.2981-3	pacific(502) security(494)	1.4434	10
8.2896-3	present(496) intervention(503)	1.4521	10
8.2868-3	go(500) go(500)	1.4551	10
8.2585-3	bills(505) holdings(505)	1.4843	10
8.2105-3	cents(599) barrel(440)	1.5337	10

Table 2.2: The 5 best- and worst-ranked bigrams of frequency 10

In contrast to this high-ranked 4-gram, the last-ranked 4-gram occupies the $3,508^{th}$ position: *"issuing indicated par europe"* with t-test value 0.009698, observed frequency 10, and expected frequency $22.25e-07$.

In our earlier work, we compared our ranking, based on the expected document frequency of discontiguous word sequences to another ranking obtained through a well-known technique.

Unfortunately, such a "ranking comparison" could only be empirical, since our standpoint was to focus on general-purpose descriptors. It is therefore, by definition, impossible to assess descriptors individually as interesting and not.

We therefore ranked word pairs through pointwise mutual information. We decided to opt for pointwise mutual information [Fan61] as applied to collocation discovery by Church and Hanks [CH90].

The rank of all word pairs is obtained by comparing the frequency of each pair to the probability that both words occur together by chance. Given the independence assumption, the probability that two words occur together by chance is the multiplication of the probability of occurrence of each word. And pointwise mutual

Bigram	Frequency	Mutual Information
het(11) comite(10)	10	17.872
corpus(12) christi(12)	12	17.747
kuala(14) lumpur(13)	13	17.524
piper(14) jaffray(10)	10	17.524
cavaco(15) silva(11)	11	17.425
lazard(16) freres(16)	16	17.332
macmillan(16) bloedel(13)	13	17.332
tadashi(11) kuranari(16)	11	17.332
hoare(15) govett(14)	13	17.318
ortiz(16) mena(14)	13	17.225

Table 2.3: Mutual Information: the 10 best bigrams.

Bigram	Frequency	Mutual Information
het(11) comite(10)	10	17.872
piper(14) jaffray(10)	10	17.524
wildlife(18) refuge(10)	10	17.162
tate(14) lyle(14)	10	17.039
g.d(10) searle(20)	10	17.010
pacific(502) security(494)	10	6.734
present(496) intervention(503)	10	6.725
go(500) go(500)	10	6.722
bills(505) holdings(505)	10	6.693
cents(599) barrel(440)	10	6.646

Table 2.4: Mutual Information: the 5 best and worst bigrams of frequency 10.

information is thus calculated as follows:

$$I(w_1, w_2) = log_2 \frac{P(w_1, w_2)}{P(w_1)P(w_2)}.$$

If $I(w_1, w_2)$ is positive, and thus $P(w_1, w_2)$ is greater than $P(w_1)P(w_2)$, it means than the words w_1 and w_2 occur together more frequently than chance. In practice, the mutual information of all the pairs is greater than zero, due to the fact that the maximal frequent sequences that we want to evaluate are already a selection of statistically remarkable phrases.

As stated by Fano [Fan61], the intrinsic definition of mutual information is only valid for bigrams. Table 2.3 presents the best 10 bigrams, ranked by decreasing mutual information. Table 2.4 shows the 5 best- and worst-ranked bigrams of frequency 10 (again, the worst ranked bigrams of frequency 10 are also the worst ranked overall). We can observe that, for the same frequency, the rankings are very comparable. Where our technique outperforms mutual information is in ranking together bigrams of different frequencies. It is actually a common criticism against mutual information, to point out that the score of the lowest frequency pair is always higher,

with other things equal [MS99]. For example, the three best-ranked MFS in our evaluation, *"Los Angeles"*, *"Kiichi Miyazawa"* and *"Kidder Peabody"*, which are among the most frequent pairs, rank only 191^{st}, 261^{st} and 142^{nd} with mutual information (out of $4,038$ pairs).

Mutual information is not defined for n-grams of a size longer than two. Other techniques are defined, but they usually give much higher scores to longer n-grams, and in practice, rankings are successions of decreasing size-wise sub-rankings. A noticeable exception is the measure of mutual expectation [DGBPL00].

Compared to the state of the art, the ability to evaluate n-grams of different sizes on the same scale is one of the major strengths of our technique. Word sequences of different size are ranked together, and furthermore, the variance in their rankings is wide. While most of the descriptors are bigrams ($4,038$ out of $4,855$), the 604 trigrams are ranked between the 38^{th} and $3,721^{st}$ overall positions. For the 141 4-grams, the position range is $10-3,508$.

2.2.4 Conclusion and Perspectives

We presented a novel technique for calculating the probability and expected document frequency of any given non-contiguous lexical cohesive relation. We first calculated an exact formula to reach this result, and observed that it is not usable in practice, because of an exponential computational complexity. We then found a Markov representation of the problem and exploited the specificities of that representation to reach linear computational complexity. The initial order of complexity of $O(ln^{l-n})$ was brought down to $O(ln)$, from exponential to linear.

We further described a method that compares observed and expected document frequencies through a statistical test as a way to give a direct numerical evaluation of the intrinsic quality of a multiword unit (or of a set of multiword units). This technique does not require the work of a human expert, and it is fully language- and application-independent. It permits to efficiently compare n-grams of different length on the same scale.

A weakness that our approach shares with most language models is the assumption that terms occur independently from each other. In the future, we hope to present more advanced Markov representations that will permit to account for term dependency.

2.3 Extracting Multilingual Information

Information Extraction (IE) is a problem area in Natural Language Processing (NLP), which concerns methods for transforming information found in plain, natural-language text – such as news articles or web pages – into a structured representation – such as a database table or a spreadsheet.

Information retrieval and extraction in the medical domain is a very active field. Text mining recently emerged at the interface between natural language processing (mostly in English) and data mining techniques. While the bulk of biomedical text mining focused on academic articles for medical researchers' needs, research also started in the 1990's to cater to a wider audience, practitioners and health organisms.

Monitoring the web to detect information on epidemics and health problems is one of those applications called epidemic surveillance.

The work described in this section is the fruit of the doctoral work of Gaël Lejeune, under the joint supervision of Nadine Lucas and myself.

Epidemic surveillance Automated news surveillance is an important application of information extraction. The detection of terrorist events and economic surveillance were the first applications, in particular in the framework of the evaluation campaigns of the Message Understanding Conference (MUC). In MUC-3 [muc91] and MUC-4 [muc92], about terrorism in Latin American countries, the task of participants was, given a collection of news feed data, to fill in a predetermined semantic template containing the name of the terrorist group that perpetrated a terrorist event, the name of the victim(s), the type of event, and the date and location where it occurred. In economic surveillance, one can for instance extract mergers or corporate management changes.

An application of information extraction that lately gained much importance is that of epidemiological surveillance, with a special emphasis on the detection of disease outbreaks. Given news data, the task is to detect epidemiological events, and extract the location where they occurred, the name of the disease, the number of victims, and eventually the "case", that is, a text description of the event, that may be the "status" of victims (sick, injured, dead, hospitalised ...) or a written description of symptoms. Epidemiological surveillance has become a crucial tool with increasing world travel and the latest crises of SARS, avian flu, H1N1 ...

The work we will describe in this section presents an application to epidemic surveillance, but it may be comparably applied to any subdomain of news surveillance.

Multilingual Information Extraction As in many fields of NLP, most of the work in information extraction long focused on English data [EBSW08]. Multilingual has often been understood as adding many monolingual systems, except in pioneer multilingual parsing [Ver02]. Whereas English is nowadays the *lingua franca* in many fields (in particular, business), we will see that for several applications, this is not sufficient. Most news agencies are translating part of their feed into English (e.g., AFP[1] and Xinhua[2] for which the source languages are respectively French and Chinese), but a good deal of the data is never translated, while for the part that is, the translation process naturally incurs a delay. This is, by essence, problematic in a field where early detection and exhaustivity are crucial aspects.

Subsequently, the ability to simultaneously handle documents written in different languages is becoming a more and more important feature [BPSY08, GKK09]. Indeed, in the field of epidemiological surveillance, it is especially important to detect a new event the very first time it is mentioned, and this very first occurrence will almost always happen in the local language. Therefore, it is not enough to be able to deal with several languages : It is necessary to handle many. For instance, the Medical Information System (Medisys[3]) of the European Community gathers

[1] *Agence France Presse*, `http://www.afp.com/afpcom/en/`

[2] *Xinhua*, `http://www.xinhuanet.com/english/`

[3] Medisys, `http://medusa.jrc.it/medisys/aboutMediSys.html`

news data in 45 different languages [AVdG09].

The generic IE architecture [Hob93] with components for each linguistic layer (morphology, syntax, semantics) has proved its high efficiency for applications in some important languages. But most of the components involved are distinct for each new language, and for many of them, some of the components simply do not exist.

Following Hobbs' generic IE chain too closely means that when one wants to improve a system by dealing with a new language, one has to find or to build most of the components that form its basis. It does not seem to raise objections when efficient components are already available, or at least when one can find them in another language which has common properties [EFC+11]. A system built for Spanish might not be so difficult to modify for processing French.

The sequential aspect of an IE chain where each node depends on the results of the previous one, might provoke cascading errors [McC06]. An important drawback is that the end-user might want to process a real multilingual corpus with a lot of languages, while efficient components lack for many of them [Ste11].

Hence, in a multilingual setting, we want to avoid a cumulative process where the only way to improve a system's coverage is one language after the other. Machine learning [NÓ4] or emerging patterns [McC06] are used to limit the cost of this explosion of resources in each language. But this is not sufficient for dealing with languages for which training data is missing. Therefore factorization shall occur at the multilingual level, using language-independent properties. One such property is the way a text, in our case a press article, is structured [Luc12].

2.3.1 Multilingual IE based on Discourse Features

Main differences between languages are found at fine grain: morphemes and phonemes are difficult or impossible to compare. This is something well known when one has to deal with foreign languages. But at higher grain, the differences disappear. Almost every language community has its scientific articles, its press articles. Each genre is constructed along the same rules. Journalists have a particular way to present what linguists call topic and comment [Lam96]. When human beings have to convey a message they might use different local forms but use the same global structure (i.e., rhetorical principles). Our system, described in the next section, focuses on the global structure as opposed to the usual separate linguistic layers (e.g., morphology, syntax and semantics).

Towards discourse based extraction. The original idea presented in this paper is that information can be detected at the discourse level. Discourse properties in the news genre are the basis upon which discourse processing is conducted [Luc12]. Reporters use position and repetition to ensure safe transmission. For Dor [Dor03], the headline is a "relevance optimizer". Sensidoni showed how the main characteristics of an event, of any kind, are very explicitly expressed by the author [Sen11].

Following these ideas, the idea sprung to use a predefined "document template" in a top-down approach, to restrict the investigation domain. These templates are quite simple: what is important is described in the top part of the document and repeated in the rest of the document, as we proposed in the beginning of Gaël Lejeune's work,

in collaboration with researchers of the PULS system of the University of Helsinki, whom Gaël visited for 4 months in 2010 [19]. Following his visit, the PULS system integrated further experiments focusing on the header of documents, that is, title and first paragraph, or first n sentences [SHY11].

Language-dependent modules needed in classical approaches can be POS taggers or syntactic analyzers. The discourse-based approach allows a shortcut between input and output, avoiding the dependency on numerous processing steps and their potential cumulative errors.

Discourse-based approaches have been seldom used for morphologically rich languages where very specific tools need to be developed for words or lemmas identification [STF11]. As we will see, the generality of our system permits to address such languages without specific processing.

2.3.2 The *DAnIEL* Surveillance System

In this section, we will describe our system, named *DAnIEL*[4]: *Data Analysis for Information Extraction in any Language*.

The system represents a full discourse-level IE approach. Its aim is to extract epidemic events, in the form of "disease-location" pairs (e.g., what disease occurred in what country?). *DAnIEL* requires a small knowledge base, and its processing pipeline contains three main steps, described below: 1) Article segmentation, 2) Motifs extraction and filtering and 3) Event detection.

Knowledge base. Contrary to state of the art systems, the lexicon needed with discourse-based IE is quite small: roughly hundreds of items instead of tens of thousands [CKJ+06]. *DAnIEL* uses only light resources collected from Wikipedia with light human moderation. Therefore, it becomes possible to deal with new languages efficiently, even without the help of a native speaker. The smallness of the lexicon might be expected to impair recall very much, but according to our experiments, the number of disease terms which the system has access to is sufficient when working at text level. General terms known by newspapers readers and used by journalists are also used on the Wikipedia, and the addition of the disease lexicon of a new language can be achieved easily, by using the content of Wikipedia's interlingual hyperlinks as translations.

2.3.2.1 Article Segmentation

In the aforementioned article [19], in accordance with the press article genre, we made the assumption that the central focus of a news article was bound to be repeated at least once in two different zones of the article defined as the *header* (title and first paragraph) and the *body* (rest of the text). This is how we preselected candidate diseases and locations from any given input text.

While this functioned surprisingly well, we noticed distinct impact for articles of different sizes, and adapted the technique according to three different types of articles, simply recognized by their relative size, which, in a strongly calibrated field such a journalism, gives good enough hints about their type:

[4] *DAnIEL*, `http://www.danieltool.info/`

- short articles (3 paragraphs or less): dispatches, breaking news,
- medium articles (4 to 10 paragraphs): regular articles, event evolution,
- long articles (10 paragraphs or more): analysis articles, less current events.

Accordingly, we adapted our segmentation to these different types of documents. The size of short documents implies that their structure is less pronounced, because the marginal cognitive cost of reading the whole document versus reading only the header is small. Hence, and we decided to simply look for repetitions anywhere within the document. We kept our method unchanged for medium-sized articles, but we adapted it for longer articles, which are usually the place of deeper analysis, and normally concluded with a summary. Following, we kept looking for information in the *head* of the document, but ignored the rest of the document except for its *tail* (last paragraph).

The result of this first step is the selection of text areas within which we will look to detect events:

- For short articles: the whole document,
- For medium articles: *head* (title and first paragraph) and *body* (everything else),
- For long articles: *head* (title and first paragraph) and *tail* (last paragraph).

2.3.2.2 Motifs Extraction

This section describes the extraction of repeated motifs. To make sure we can take morphological variation and compound words into account, we extract long repeated sequences of *characters* rather than words.

This character level analysis was done with non-gapped motifs (hereafter motifs) as following the algorithm described by Ukkonen [Ukk09]. Those motifs are substrings patterns with two main characteristics: 1) they are repeated (they occur twice or more), and 2) they are maximal (they cannot be expanded to the left or to the right without reducing their frequency).

The number of different motifs is less than the number of characters in the text and they are detected in time $O(n)$ using augmented suffix arrays [KSB06].

Motifs filtering.

First, the motifs that are not repeated in the right document parts are discarded. Then, the extracted motifs are compared to the disease name lexicon to check whether the document is relevant (i.e., whether it describes an event).

Discourse-based filtering. Motifs that did not appear in each of the appropriate document segments are discarded. This discourse-based selection permits to filter out more than 85% of the detected motifs. Its precision is surprisingly good, since the analysis of a sample of the correspondingly rejected documents showed that 99% of these were irrelevant indeed.

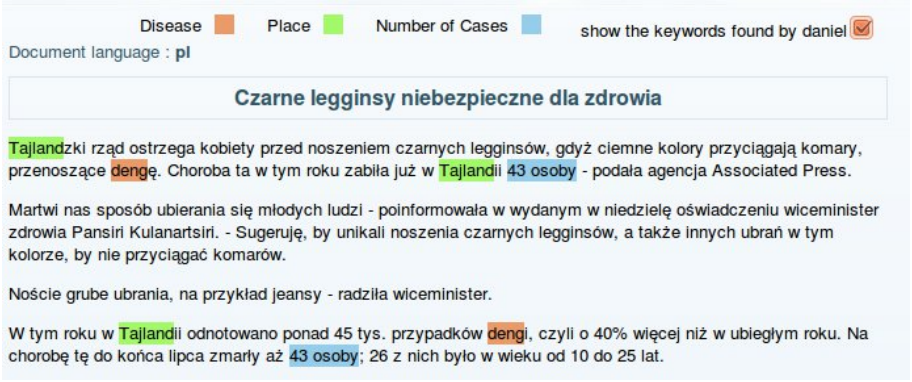

Figure 2.6: Example extraction in a medium-sized article in Polish. The system highlighted matching motifs.

Knowledge-based filtering. The remaining motifs are compared to a list of diseases. While motif extraction at the character level allows to take linguistic variation into account, such a variation needs to be taken into account within the disease list as well, if we are to match motifs and disease names.

To do this, we relied on loose matching and a heuristic ratio, deciding that a positive match was found if $\frac{3}{4}$ of the characters were common to both a motif and a disease name. This way, we avoid the need for morphological analysis.

Figure 2.6 shows an example of a Polish document analysed by *DAnIEL*, which will be fully explained in the next section. However, we can already see, in this example, that the loose matching permits to characterize that the string "deng∼" is useful eventhough the actual lemmatized form "denga", the only form available in our knowledge-base, is not present in that particular text.

2.3.2.3 Full Event Detection

Now that the disease was detected, it remains to detect the corresponding location, and further optional information about the event, known as the case.

Locating an event. Previous research has shown that learning vocabulary for geo-parsing is far from trivial even when dealing with only one language [KFB09]. Therefore, instead of adding one problem to another, we chose to restrict ourselves to the identification of the country of occurrence of the event. Defining such a fixed geographical grain further facilitates the evaluation procedure, since we do not have to ponder whether, for instance, detecting the location "Paris" is correct or incorrect when the expected answer was "Montmartre".

To locate events, we rely on the motifs extracted previously, that we compare to our knowledge-base of countries, using the exact same procedure for the loose matching of motifs and countries.

One important specificity of location, is however that it is often implicit. In situations when the journalist does not mention any location in the document, it usually means that the event occurred in the place of issue. Hence, should no other information be given, the location of the source (e.g., the newspaper) and the location of the event are considered the same.

Detecting the case. Optionally, we try to extract additional information, known as the "case", that is, additional information about the event, such as the number of victims or their status. Our heuristic is to search for a repeated motif starting with a number.

Example extracted event in Polish. A sample detected event is given in Figure 2.6. The (disease, location) pair is present in the first and fourth paragraphs, and so is the extracted case: "43 osoby" (43 victims). The motif "deng∼" is repeated in two different forms and sometimes referred to by the more general term "chorob∼" (disease). The name of the country involved, "Tajland∼" is also found both in the *head* and *body* parts. Hence, the event is first expressed and then developed, which is a very typical expression of the topic/comment structure described by Lambrecht [Lam96]. Interestingly, we can see in this example that the title does not mention the event directly, as it focuses on a contextual aspect; that wearing leggings increases the risk of being stung by a bug carrying the dengue virus.

Example extracted events in Russian and Arabic. To underline the language-independence of the approach, we are presenting hereby two further examples in Russian and Arabic. In the Russian article given in Figure 2.7, the disease identified is swine flu, and the location is "worldwide". The extracted case, number of victims of the epidemics, is 4379. When it comes to the Arabic article (Figure 2.8), even without any personal knowledge in Arabic, I can tell with strong confidence that an epidemiological event was detected, and I can point at the name of the disease. In this article, the location is mentioned only once, in what seems like a final copyright line. It probably does not differ from the location of the source, which is what *DAnIEL* will use as the location of this event, since the detected location is not repeated.

Many other examples are available on *DAnIEL*'s dedicated Web site[5], where many more articles shall be found, in various languages.

2.3.2.4 Current Results

For several languages, Table 2.5 shows the precision, recall and F-measure, as well as the size of the evaluation sample.

The main problem we face in evaluating this work is to compare it to that of others, since state of the art systems either deal with other application domains than epidemiology, or are not sharing any of their annotation data. And, should they be willing to share their data, it would cover only a few languages.

Hence, to get a first glimpse at where our system stands compared to others, we shall compare its results to those reported on data sets that are comparable. In this sense, state of that art for English is reached by BioCaster's Global Health Monitor with 93% precision [CKJ+06]. Our system reaches 84%, which, considering difference in the amount of resources involved is actually very encouraging.

In addition, we should underline once more that we are reporting results for several languages that state of the art systems do not deal with. This means that,

[5] `http://www.danieltool.info/`

Figure 2.7: Example extraction in a medium-sized article in Russian. The system highlighted matching motifs.

Figure 2.8: Example extraction in a medium-sized article in Arabic. The system highlighted matching motifs.

	Recall	Precision	F-measure	Documents
English	88%	84%	86%	200
Chinese	92%	85%	88%	100
French	92%	84%	88%	1,954
Greek	89%	83%	86%	274
Polish	85%	83%	84%	283
Russian	89%	83%	86%	305
Spanish	91%	83%	87%	120

Table 2.5: Performance of *DAnIEL* for event detection, and number of annotated documents.

even if *DAnIEL* is 9 points behind in terms of precision for the English language, we could say that it is 83 points ahead for the Polish language, with 83% precision.

Evaluation. Since no annotated data is publicly available, we decided to collect our own, and distribute it freely. *DAnIEL*'s online platform lets visitors annotate documents freely, and gathers collected data for evaluation.

Opening an annotation platform appeared as the only way to collect annotations in nearly any language. So far, we have invited identified users to annotate different sets of documents, corresponding to their language skills.

In the future, we hope to open the platform and let people assess documents as they browse through the interface. Ordering crowdsourcing batches is also on the agenda, once protection against fraudulent annotations is in place. Up to this point, we only trust annotations from registered and validated users.

For the Russian, Polish and Greek documents, inter-annotator agreement computed with Fleiss's kappa, is 81%, suggesting that the annotation task is sufficiently clear to the annotators.

2.3.3 Conclusion and Perspectives

This section introduced *DAnIEL*, a discourse-based information extraction system applied to the detection of epidemiological events. It is intended to help health authorities get precious information about ongoing infectious diseases spreading all around the world. In order to be multilingual, it uses as much factorization as possible and relies on text-level rather than sentence-level.

It is based on the way that press articles are constructed and on how human readers decode documents by skimming over them. The detection of string repetitions simulates this behaviour. Communication principles are used rather than usual linguistic layer analysis. This permits to limit the number of components needed for monitoring new languages. No local analysis is used and a limited-size lexicon is enough for efficient analysis.

With an average precision of 83.4%, *DAnIEL* performs only slightly worse than state of the art systems like PULS [SFvdG+08] or Global Health Monitor [CKJ+06] which are closer to 90% on English and a few other languages. But the resources that these systems need (language analyzer, lexicon or ontologies) are much heavier. Our approach can then be a good substitute for a state of the art IE system when

resources are scarce. It can also be an efficient addition, to handle all the languages that language-focused systems cannot process.

Low resources. *DAnIEL* requires resources in a scale of 100 times less than state of the art methods. For instance, PULS needs tens of thousands of entries, consisting essentially of language-specific lexicons and grammatical extraction patterns. *DAnIEL*, to add any language, only requires a list of about 200–300 diseases, and a similar-sized list of country names. For most languages, these small resources are fairly easy to gather automatically.

Efficient processing. A consequence of the small amount of resources, and of the linear motif extraction algorithm, run only on parts of the documents, implies very efficient processing. *DAnIEL*, written in PYTHON, can indeed process 2,000 documents in less than 45 seconds on a 2.4 Ghz processor with 2Gb RAM. This is about 10 times faster than the PULS system.

***DAnIEL* platform for the collection and distribution of annotations.** To allow for the comparative evaluation of surveillance systems in a multilingual framework, we have created the *DAnIEL* platform, where users can browse through documents and alerts of the *DAnIEL* system, and can annotate the documents by signaling an event. It was developed in 2011 by master's student Benoit Samson, now graduated.

In order to help IE research, notably for low-resource languages, all the annotations gathered through the *DAnIEL* platform are available to the community.

Perspectives

In this line of research, I foresee two main lines of work. The first one concerns an essential strengthening of the evaluation of our current research. The second one is a generalisation of the approach to other domains than epidemic surveillance.

Crowdsourcing. When evaluating this research, we are facing several problems. Since we do not want to restrict the list of languages that the system can handle, we have no reason to restrict the list of languages to be evaluated. But while gathering a sufficient amount of annotations is generally difficult, it is even more so when the amount of annotations needs to be multiplied by the number of languages to be evaluated... In addition, finding competent and willing annotators for any language is a grueling and hardly realistic task.

A potential solution to these problems is to rely on crowdsourcing. For the moment, only authorized users are allowed to annotate the documents in *DAnIEL*, but in the future, we plan to open its access to crowdsourcing via, e.g., Amazon's MTurk. Through my research in evaluation, I have acquired experience in setting up crowdsourcing methodologies (see Chapter 4), which would be an ideal solution for acquiring annotations in a wide range of languages, about most of which we have no competence at all. This will require setting up solid procedures to avoid fraudulent annotations.

Generalisation to other application domains... without resources? After reducing the size of external resources from tens of thousands entries to hundreds, the next step shall be to try to work without any external resources.

In addition, the performance obtained with epidemiological surveillance can probably be transposed to other domain, such as, e.g., finance. However, doing so should ideally be done efficiently.

Following my temporary position as a full time CNRS researcher in 2011-2012, I stayed at the University of Helsinki where we started a collaboration with Hannu Toivonen. Based on the idea of cognitive mapping, we are working on a technique to detect novel relationships in documents entering a news stream. Such a detection stems from the combination and comparison of *local* association measures (built from the incoming document) and *global* association measures (built from the collection of previous documents).

The aim of such a "news novelty detector" is to be able, for instance, to automatically detect the surprising co-occurrence of "Dengue" and "Thailand", or "Apple Inc." and "profit warning".

The system would not need to be designed for a specific application domain, but the user would rather fine-tune which application domains she is interested in, in her personal interface.

Preliminary experiments were already led with the University of Helsinki and promising results were obtained with sentence-based statistics (with a $tpf-idf$ measure: combination of Term Pair Frequency (tpf) and Inverse Document Frequency). However the results are very noisy, and while the approach maintains domain- and language-independence, these early experiments were performed at the word level, which should cause severe difficulties with morphologically rich languages.

2.4 Related Publications

Extraction and selection of sequential data from text. The first two sections dealt with my doctoral work on sequential descriptors, and corresponding material is evidently found in my Ph.D. thesis [36, 37]. Two papers are more specifically dealing with the extraction technique [1, 16], while the linear time computation of the probability of discontinued occurrence of a sequence was fully described in the TAL journal in 2006 [9]. A preliminary version of this work was presented in a SIGIR workshop in 2005 [29]. More recent research described a tentative generalization of mutual information to sequences of length more than two. This latest work was presented at the 2010 international conference on Knowledge Discovery and Information Retrieval (KDIR) [12].

Numerous other personal publications are dealing with applications of the extracted sequences [3, 4, 14, 17, 23, 24, 26, 28, 30, 31, 33]. These applications are described in Chapter 3 of the present manuscript, dealing specifically with the ways to make use of extracted knowledge.

Multilingual information extraction. The main ideas of this work were first exposed in the MinUCS workshop [21], and later, in French, in the 2010 "Journées internationales d'Analyse statistique des Données Textuelles" (JADT) [11]. These

first two papers presented language-independent techniques to integrate new languages into an existing system. This first step in this line of work was test-cased with the integration of French and Spanish within the PULS monitoring system of the University of Helsinki. This work took place within the CNRS MultiPULS PICS project, and through a 4-month internship of Gaël Lejeune within Roman Yangarber's research group in Helsinki.

Further experiments showed that the technique permitted reaching performance close to that of state of the art methods on English, French and Chinese. They were published within the 4^{th} Cross-Lingual Information Access (CLIA) workshop co-located with the international Conference on Computational Linguistics (COLING) 2010 [19].

Finally, the *DAnIEL* surveillance system is extensively described, together with results on Greek, Russian and Polish, in a paper entitled "*DAnIEL*: a Multilingual Surveillance System based on Discourse Features", recently submitted to the annual meeting of the Association for Computational Linguistics (ACL 2012).

2.5 Conclusion

In this chapter on the extraction of knowledge, we have presented research on the use of the sequential nature of text for document representations on one side, and a low-resource set up for multilingual information extraction on the other. In both cases, we made the voluntary choice of generality. We developed techniques that are suited for general document collections written in any language.

The first result is the development of an efficient technique for the extraction of a compact set of word sequences from text. Built upon *MineMFS*, an existing technique for the extraction of maximal frequent sequences, we presented *MFS_MineSweep*, an improvement of this technique that is based on splitting the original document collection into homogeneous partitions. The outcome is that we can obtain more exhaustive results faster. Further, a drawback of *MineMFS* is that it fails to produce any descriptor at all for large document collections, whereas our contribution permits to extract phrasal descriptors out of a collection of virtually any size.

The following result permitted filling a gap in the current research on multiword units and their use for information retrieval applications. We presented efficient algorithms for computing the probability and expected frequency of occurrence of a given sequence of items. One application of this technique is the direct evaluation of sequences, notably word sequences, obtained by interestingness measures that are calculated by comparing the expected and observed document frequencies of a sequence. The more the hypothesis that a sequence occurs by pure chance is wrong, the more that sequence is interesting with respect to the corpus. Our technique offers an efficient alternative to the current evaluation methods of word sequences. These techniques are indeed essentially task-based, relying on time-consuming manual assessments, as is the case in lexicography, or embedded within an application framework as is usually done in information retrieval. The weakness of indirect evaluation is that it remains difficult to decide whether any result stems from the quality of the phrases or from the way they were used. The evident benefit of a direct evaluation technique such as ours is that its results are easier to interpret, as

neither intervenes a separate application, nor a subjective human judgment.

Our last main contribution to the extraction of knowledge stems from the design of a general approach for the automatic detection of epidemiological events in a multilingual setting. Adding additional languages is very straightforward, since it requires only small lexicons that can easily be obtained automatically. As opposed to state of the art systems, which rely on thousands or tens of thousands of language-specific patterns, this is a major improvement. Such a system is faster to implement, and much faster to execute. Its performance is close to that of state of the art techniques in the languages they handle (i.e. usually, English, French or Chinese). Our approach is by definition infinitely better-performing for languages that other systems do not deal with: Polish, Russian, Greek, Finnish, Arabic, ... and French or Chinese for those systems that do not deal with them. The only language that is common to nearly all state of the art systems is English, one of the most simple languages for automatic processing, thanks notably to its isolating nature and very limited inflections. In other words, English is especially suited for automated systems, and reciprocally, automated systems are especially suited for English. Hence, the dominance of English as a global language is perhaps not the only reason why state of the art techniques seldom address any other language, and do particularly ignore low-resource languages.

Chapter 3

Exploiting Knowledge

The extraction of knowledge is generally not sufficient in itself. While multiword units can, as such, satisfy a lexicographer, knowledge extraction is most often a first step towards practical applications.

In this chapter, I will introduce experiments in four distinct fields of research. Grossly, the first two sections deal with applications of the extraction of MFSs, while the last two sections deal with structured documents. However, we will see that these research activities are interleaved, under the general principles of being equally usable, whatever the language of the corpus, and whatever the type of document. When it comes to structured documents, this does not only mean that any genre or domain can be addressed, but also that no DTD or schema is required, and that all the techniques will function even if documents are not well-formed.

In Section 3.1, we present a general method to calculate the phrase-based similarity of documents, that we have applied to information retrieval. The language- and domain-independence of the approach is underlined by experiments in Japanese, Chinese, Korean and English, on scientific and news feed articles.

In Section 3.2, we present the outcome of a collaboration with the University of Beira Interior (Portugal), in which MFSs have proved particularly useful. The global aim of this joint work was to discover word semantic relations by observing variations of vocabulary in sets of paraphrases. Our main assumption was that the occurrence of two words (or word groups) within a similar paraphrase context was a strong indicator of a semantic relation between those words. We used the MFSs of a set of paraphrases as a back-bone, around which the paraphrase were aligned, so as to emphasize their variations. To these means, we designed a method that can achieve multiple sequence alignment in one pass, after the MFS set was extracted.

The next two sections deal with the exploitation of structured information. In Section 3.3, we introduce the EXTIRP structured information retrieval system, developed in 2003 at the University of Helsinki, and used yearly within the INEX evaluation initiative until 2009. It addressed the problem of granularity in information retrieval by defining minimal retrieval units at lower levels of the document tree, before propagating relevance values in a bottom-up fashion, to finally decide on the optimal document fragments to be returned to a query. EXTIRP has been the framework of several research works within the Doremi group[1] during that period

[1] DOcument management, information REtrieval, and text and data MIning, `http://www.cs.helsinki.fi/group/doremi/`

of time, including the doctoral work of Miro Lehtonen, who graduated in 2006. My collaboration with Miro was fruitful and led for instance to the successful integration of structural features into the MFS extraction process.

The last application presented in this chapter is that of the non-supervised classification of XML documents. We proposed to rely on structural features to efficiently distinguish outliers, before performing "regular" clustering (Section 3.4).

To conclude this chapter focused on applications, Section 3.5 gives a contextualized summary of my related publications.

3.1 Computation of Phrase-Based Similarities

As opposed to words, the higher content specificity of phrases is a strong motivation for their extraction. The potential improvement that may be obtained by using phrases in document retrieval is supported by the behavior of users. In an analysis of the query log of the Excite search engine (more than 1.5 million queries), Williams et al. [WZB04] found that 8.4% of the queries contained explicit phrases, that is, they included at least two words enclosed in quotes. Even more interestingly, the authors found it beneficial to treat 40% of the queries without quotation marks as phrases rather than independent words. Consequently, there is no doubt that an efficient technique to use phrases may bring solid improvement to document retrieval applications. In a context such as the Web, where numerous languages coexist in enormous collections for which scaling is a key issue, it is crucial to use techniques that are language independent. All our work is entirely corpus independent (and in particular, language independent), only relying on knowledge present *inside* the document collection being processed.

3.1.1 State of the Art

Work on the use of phrases in IR has been carried out for more than 35 years. Early results were very promising. However, unexpectedly, the constant growth of test collections caused a drastic fall in performance improvements. Salton et al. [SYY75] showed a relative improvement in average precision, measured over 10 recall points, between 17% and 39%. Fagan [Fag89] reiterated the exact same experiments with a 10 Mb collection and obtained improvements from 11% to 20%. This negative impact of the collection size was later confirmed by Mitra et al. [MBSC97] over a 655 Mb collection, improving the average precision by only one percent. Turpin and Moffat [TM99] revisited and extended this work to obtain improvements between 4% and 6%.

In our opinion, this does not contradict the idea that adding document descriptors accounting for word order is likely to improve the performance of IR systems. One problem is the extraction of the phrases, while another difficult related problem is to find efficient ways to benefit from those phrases. This need was illustrated by work of Lewis [Lew92] and Vechtomova [Vec05], who both decided to involve human experts in the process. Both obtained small improvements, suggesting that the techniques to exploit the extracted phrases can also be improved.

There are various ways to exploit phrase descriptors. The most common technique is to consider phrases as supplementary terms of the vector space, using the

same technique as for word terms. In other words, phrases are thrown into the bag of words. However, according to Strzalkowski and Carballo [SC96], using a standard weighting scheme is inappropriate for mixed feature sets (such as single words and phrases). In such cases, the weight given to the least frequent phrases is considered too low. Their specificity is nevertheless often crucial in order to determine the relevance of a document, but while weighting phrasal matches, the interdependency between a phrase and its word components is another difficult issue to account for.

Vechtomova introduced an advanced matching technique [Vec05]. Its contribution was to address the problem of overlapping phrases, in a way that accounts for the relative positions of occurrence of the words they contain. The problem of overlapping phrases occurs for phrases of more than *two* words. Given a query phrase ABC, it is the question of how to evaluate a document that contains the phrase ABC and a document that contains the phrases AB and BC separately.

For each query phrase, a pass through the document collection is done, to retain every occurrence of terms of the query phrase and their original positions in the document. Terms that form the keyphrase or one of its sub-phrases are gathered into so-called "windows". Each window is weighted by the inverted document frequency (idf) of the words that compose it and the distance that separated them originally:

$$WindowWeight(w) = \sum_{i \in w} idf_i \times \frac{n}{(span+1)^p},$$

where i is a word occurring in the window w, n is the number of words in the window w, *span* is the distance between the i^{th} and last word of the window, and p is a tuning parameter, arbitrarily set to 0.2. The score attributed to each document is calculated as the sum of the weights of the phrases it contains, where the weight of a phrase a in a document is defined as follows:

$$PhraseWeight(a) = \frac{(k+1) \times \sum_{w=1}^{n} WindowWeight(w)}{k \times NF + n},$$

where n is the number of windows w extracted for the phrase a, k is a phrase frequency normalization factor, arbitrarily set to 1.2. and NF is a document length normalization factor:

$$NF = (1-b) + b \times \frac{DocLen}{AveDocLen},$$

where *DocLen* and *AveDocLen* are the document length and the average document length in the corpus (number of words), and b is a tuning constant, set to 0.75.

A major drawback is the computational complexity of this process. In this method, there is no static phrase index that gives a phrasal representation of the document collection. It is only at query-time that a representation of the collection is built that only contains the terms of the query. Such heavy processing in response to a query is quite problematic, as users usually expect to obtain results promptly.

In practice, the method has only been used for re-ranking the $1,000$ best documents returned to a query by a vector space model relying on single word features. The results demonstrate a performance improvement in terms of average precision, which is unfortunately not statistically significant. They also confirm a common observation when using phrases for document retrieval: compared to the use of single

word features only, improvement is observed at high recall levels, while the impact is negative at lower levels.

The following section introduces our technique for computing phrase-based document similarity, and for integrating it in a formal document retrieval framework.

3.1.2 Enhancing Retrieval using Phrases

Problem definition. Given a set of sequences that describe the documents of a collection, how can we determine to what extent the sequence $p_1 \dots p_n$, issued from the document collection, corresponds to the sequence $q_1 \dots q_m$, found in a user query? And how can we subsequently rank the documents according to how well we think they answer to the query?

3.1.2.1 Desired Features of Phrase Matching

We propose an approach that consists in comparing a set of descriptive phrases extracted from the document collection, to a set of *keyphrases* from the query. Given a query, every document receives a reward for every sequence it contains that matches a keyphrase of the query. This bonus generally differs for each different phrase. Note that from here onwards, the term *keyphrase* will be used to refer to a phrase found in a query.

A base weight. The most informative lexical associations should notably be promoted, using statistical information such as term and inverted document frequency.

Longer matches are better matches. Further, it is natural to wish that longer matches should receive a higher reward. If a query contains the keyphrase "XML structured information retrieval", the most appropriate documents are those whose descriptors contain this exact sequence, followed by those containing a subsequence of size 3 (e.g., "structured information retrieval"), and finally by documents containing a subpair of the keyphrase (e.g., "structured information" or "information retrieval").

Adjacency should not be required. Clearly, a phrasal descriptor containing the pair "XML retrieval" has a relationship with the keyphrase "XML structured information retrieval". This illustrates the fact that natural language is richer in variety than only recurrent adjacent word sequences.

But adjacency is generally a stronger indicator. We should, however, bear in mind the general rule that the more distant two words are, the less likely they are to be related. And the degree to which the relatedness of two words is affected by distance certainly varies greatly with different languages.

Inverted usage. An extension of the previous comments about word adjacency is that we should also try to take into account the fact that words might as well occur in inverted order, while still not necessarily being adjacent. For example, a

phrase "retrieval of XML" triggers interest with respect to the earlier keyphrase "XML structured information retrieval".

Jones and Sinclair [JS74] give the example of the pair "hard work", where throughout their document collection, the words "hard" and "work" are occurring together in arbitrary order, and with a variable distance between them. Of course, in English, not all collocations are this relaxed, and others are exclusively rigid, for example the pair "Los Angeles" is very unlikely to occur in a different order, or with other words inserted. They term those two types of collocations as *position dependent* and *position free* collocations. By attributing a positive score to matches and ignoring misses, we can get around this problem. If we look for phrasal document descriptors containing "Angeles Los" or for the occurrence of "Los" and "Angeles" separated by other words, and we fail to find any, it will not worsen the retrieval performance. Whereas finding that a document about "retrieval of XML" is relevant to a query about "XML retrieval" is evidently better than failing to observe it.

In the next subsection, we will introduce our approach to the problem. It aims at taking into account all the observations above in a sensible way.

3.1.2.2 Document Score Calculation

Our approach exploits and combines two complementary document representations. One is based on single word terms, in the vector space model, and the other is a phrasal description, taking the sequential nature of text data into account.

Once documents and queries are represented within those two models, a way to estimate the relevance of a document with respect to a query remains to be found. We must sort the document list with respect to each query, which is why we need to compute a *Retrieval Status Value (RSV)* for each document and query. Below, we will explain how we calculate two separate RSVs, one for a word features vector space model and one for our phrasal description.

The reason to compute an RSV value based on the word-feature vector space model in addition to a phrasal RSV is due to the fact that the latter may not be sufficiently discriminating. A query may for instance contain no keyphrases, and a document may be represented with no phrasal descriptor. However, there can of course be correct answers to such queries, and such documents may be relevant to some information needs. Also, all documents containing the same matching phrases get the same phrasal RSV. If the phrasal description is small, it is necessary to find a way to break ties. The cosine similarity measure based on word features is very appropriate for that.

To combine both RSVs into one single score, we must first make them comparable by mapping them to a common interval. To do so, we used *Max Norm*, as presented by Lee [Lee95], which permits to bring all positive scores within the range [0,1]:

$$New\ Score = \frac{Old\ Score}{Max\ Score}$$

Following this normalization step, we aggregate both RSVs using a linear interpolation factor λ representing the relative weight of scores obtained with each technique.

$$Aggregated\ Score = \lambda \cdot RSV_{Word\_Features} + (1 - \lambda) \cdot RSV_{Phrasal},$$

```
<Keywords>
  "concurrency control"
  "semantic transaction management"
  "application" "performance benefit"
  "prototype" "simulation" "analysis"
</Keywords>
```

Figure 3.1: Topic 47

where details on the computation of both RSVs are given in the rest of this section.

An intuitive weighting scheme showed good results during early experiments with the INEX collection [30]: weighting the single word RSV with the number of distinct word terms in the query (let a be that number), and the phrasal RSV with the number of distinct word terms found in keyphrases of the query (let b be that number). Thus:

$$\lambda = \frac{a}{a+b}$$

For example, in Figure 3.1, showing topic 47 of the INEX collection, there are 11 distinct word terms and 7 distinct word terms occurring in keyphrases. Thus, for this topic, we have $\lambda = \frac{11}{11+7} \approx 0.61$.

Word features RSV. This first document representation is a standard vector space model, of which all features are single words. It represents a baseline model that our goal is to improve by the addition of sequential information from our second document model.

The index term vocabulary W includes every word found in the document collection, without preselection. Further, the words are left in their original form, no lemmatization or stemming being performed. This guarantees generality, as this can be done in an equally simple way for document collections written in any language.

In our vector space model, each document is represented by a $\|W\|$-dimensional vector filled in with a weight standing for the importance of each word token with respect to the document. To calculate this weight, we use a term-frequency normalized version of term-weighted components, as described by Salton et al. [SB88], that is:

$$tfidf_w = \frac{tf_w \cdot \log \frac{|D|}{df_w}}{\sqrt{\sum_{w_i \in W} \left(tf_{w_i} \cdot \log \frac{|D|}{df_{w_i}} \right)^2}}$$

where tf_w and df_w are the term and document frequencies of the word w, and $|D|$ is the total number of documents in the collection D.

The vector space model offers a very convenient framework for computing similarities between documents and queries. Among the number of techniques to compare two vectors, we chose cosine similarity because of its computational efficiency. By normalizing the vectors, which we do in the indexing phase, $cosine(\overrightarrow{d_1}, \overrightarrow{d_2})$ indeed simplifies to the vector product $(d_1 \cdot d_2)$.

We have already expanded on the weaknesses and the amount of information that such a simple model cannot catch. This is why we will complement this model

with a phrasal one, bringing sequential information into the document model, and aiming to carry it on into document retrieval.

3.1.2.3 Phrasal RSV

Given a set of n-grams (keyphrases) is attached to each document, we ought to define a procedure to match a phrase describing a document and a keyphrase. Our approach consists in decomposing keyphrases of the query into key pairs. Each of these pairs is bound to a score representing its inherent *quantity of relevance*. Informally speaking, the quantity of relevance of a key pair tells how much it makes a document relevant to contain an occurrence of this pair. This value depends on a basic measure of the importance of the pair (its *base weight*, which can be its inverted document frequency, for example) combined with a number of modifiers, meant to take into account the distance between two words of a pair, to penalize their possible inverted usage, and so on.

Definitions. Let D be a document collection and $K_1 \dots K_m$ a keyphrase of size m. Let K_i and K_j be two words of $K_1 \dots K_m$. We define the quantity of relevance associated to the key pair K_iK_j as:

$$Q_{rel}(K_iK_j) = Base\_Weight(K_iK_j, D) \cdot Integrity(K_iK_j),$$

where $Base\_Weight(K_iK_j, D)$ represents the general importance of K_iK_j in the collection D. A possible measure of this kind is the statistical significance of the pair, or its specificity, measured in terms of inverted document frequency:

$$idf(K_iK_j, D) = \log\left(\frac{|D|}{df(K_iK_j)}\right),$$

Integrity Modifier. When decomposing the keyphrase $K_1 \dots K_m$ into pairs, the *Integrity Modifier* of the key pair K_iK_j is defined as the combination of a number of modifiers:

$$Integrity(K_iK_j) = adj(K_iK_j) \cdot inv(K_iK_j) \cdot dup(K_iK_j).$$

Non-adjacency penalty. $Adj(K_iK_j)$ is a score modifier meant to penalize key pairs formed from non-adjacent words. Let $d(K_i, K_j)$ be the distance between K_i and K_j, that is, the number of other words appearing in the keyphrase between K_i and K_j ($d(K_i, K_j) = 0$ means that K_i and K_j are adjacent). We define:

$$adj(K_iK_j) = \begin{cases} 1, & & \text{if } d(K_i, K_j) = 0 \\ \alpha_1, & 0 \leq \alpha_1 \leq 1, & \text{if } d(K_i, K_j) = 1 \\ \alpha_2, & 0 \leq \alpha_2 \leq \alpha_1 & \text{if } d(K_i, K_j) = 2 \\ \dots & & \\ \alpha_{m-2}, & 0 \leq \alpha_{m-2} \leq \alpha_{m-3}, & \text{if } d(K_i, K_j) = m-2 \end{cases}$$

Accordingly, the larger the distance between the two words, the lower the quantity of relevance attributed to the corresponding pair. In the experiments, we set only a base value of non-adjacency penalty *adj_pen* that is raised to the power of

the distance between the two words of the key pair. In other words, $\alpha_{d(K_i,K_j)} = adj\_pen^{d(K_i,K_j)}$. In practice, choosing the example value of 0.9 for adj_pen means that the base matching quantity awarded to documents containing K_iK_j is lowered by 10% for every other word occurring between K_i and K_j in the original keyphrase.

A further possibility is to define a maximal distance between two words by setting, for example, $\alpha_k = 0$, for k greater than a given maximal distance threshold. A maximal distance of 5 was suggested for English document collections. Jones and Sinclair indeed showed that no two English words are linguistically connected if they are separated by more than 5 other words [JS74].

Inversion penalty. $Inv(K_iK_j)$ is another score modifier used to penalize key pairs K_iK_j that occur in the opposite order in the original keyphrase:

$$inv(K_iK_j) = \begin{cases} 1, & \text{if } K_i \text{ occurs before } K_j. \\ inv\_pen \leq 1, & \text{otherwise.} \end{cases}$$

Clearly, the non-adjacency and inversion penalties are strongly language- and domain-dependent. The less relative word positions matter, the lower those penalties should be. For a theoretical document collection where relative word positions have no importance, we should have $inv\_pen = 1$ and, for $0 \leq l \leq (m-2)$, $\alpha_l = 1$.

Duplication bonus. A result of the creation of non-adjacent and inverted key pairs is that, whenever one word occurs more than once in a query, the list of word pairs representing the query may contain duplicates. Rather than incrementing a corresponding number of matching quantities, we decide to remove the duplicates, and keep one occurrence of the key pair together with its highest associated matching quantity. This highest matching quantity is further increased by $dup(K_iK_j)$, a relative weight increase awarded to those pairs occurring several times in the original keyphrase.

Maximal matching distance. Observe that the question of which parts of a document descriptor can be matched with a pair was left open. If the phrasal descriptors are maximal frequent sequences, it is a sensible option to allow for an unlimited gap between each two words of the descriptor, because by definition, if $ABCD$ is frequent, then so are AB, AC, AD, BC, BD, and CD. In the general case, however, we allow for the possibility to use a maximal matching distance max_d. We try to match two words of a phrasal descriptor against a key pair only if there are no more than max_d other words occurring between them.

Example. To illustrate these definitions, let us have a look at the decomposition of the keyphrase $ABCD$. It is decomposed into 12 tuples (pair, integrity modifier):

$(AB, 1)$, (AC, α_1), (AD, α_2),$(BC, 1)$, (BD, α_1), $(CD, 1)$,
(BA, inv_pen), $(CA, \alpha_1 \cdot inv\_pen)$, $(DA, \alpha_2 \cdot inv\_pen)$,
(CB, inv_pen), $(DB, \alpha_1 \cdot inv\_pen)$, (DC, inv_pen).

Let us compare this keyphrase to the documents d_1, d_2, d_3, d_4 and d_5, represented respectively by the phrasal descriptors AB, ACD, AFB, ABC and ACB. The

Document	Description	Quantity of relevance
d_1	AB	$Bw(AB)$
d_2	ACD	$Bw(CD) + \alpha_1 Bw(AC) + \alpha_2 Bw(AD)$
d_3	AFB	$Bw(AB)$
d_4	ABC	$Bw(AB) + Bw(BC) + \alpha_1 Bw(AC)$
d_5	ACB	$Bw(AB) + \alpha_1 Bw(AC) + \alpha_1 \cdot inv\_pen \cdot Bw(CB)$

Table 3.1: Quantity of relevance stemming from various indexing phrases with respect to a keyphrase query $ABCD$. Bw stands for $Base\_Weight$.

maximal matching distance max_d is set higher than 1. The corresponding quantities of relevance brought by each matching subpart of the keyphrase $ABCD$ are shown in Table 3.1.

Assuming equal $Base\_Weight$ values, we observe that the quantities of relevance form an order matching the desirable properties that we had wished for in Section 3.1.2.1. The longest matches rank first, and matches of equal size are untied by relative word positions (adjacency and inversion). Moreover, non-adjacent matches (AC and ABC) are taken into account, unlike in many other phrase representations [MBSC97].

3.1.3 Document Retrieval Experiments

We will discuss our evaluation of a set of phrases as content descriptors in the application domain of document retrieval.

A brief reminder on evaluation measures in document retrieval. The effectiveness of a document retrieval system is measured by comparing the document ranking it generates to the set of *relevance assessments*, a list of the documents of the collection that were judged as relevant and not by domain experts.

Precision measures the proportion of relevant answers among those submitted. Recall measure the relative number of relevant documents found. Since those two measures are interdependent, evaluation methods are generally based on a combination of these two measures. An approach to estimate the quality of a list of retrieved documents is to plot a *recall-precision graph*. The graph is drawn by extrapolation from a number of data points. Typical data points are measures of precision at every 10% of recall, i.e., at recall 0, 0.1, 0.2,..., and 1. For example, the precision at recall 0.4 measures the proportion of all documents the user has to go through in order to find 40% of the relevant documents.

A subsequent popular measure of the quality of a ranked list of documents is the *average precision* over a number of points of recall. For example, for the data points at every 10% of recall, we talk about 11-*point average precision*. Reading the ranked document list from top to bottom, we can also calculate the precision each time a true positive is encountered. By averaging all those precision values together for one query, we obtain a popular measure, the *average precision (AP)*. The *mean average precision (MAP)* is the average of AP across all the queries of a test set. It is central to the evaluation of this work.

Language	Documents	Topics
Chinese	381,681	42
Japanese	220,078	42
Korean	66,146	30
English	22,927	30
INEX (en)	12,107	30

Table 3.2: Number of documents and fully assessed topics in the NTCIR-3 and INEX collections, per language.

Document collections. To evaluate the performance of our phrase-based similarity measure, we performed document retrieval experiments. Since one strength of the approach is that it is entirely language- and domain-independent, we ran our experiments with document collections of different domains and written in different languages.

We hence experimented with the NTCIR-3 collection[2], containing news-feed documents in four distinct languages, namely, English, Japanese, Chinese and Korean. Since Chinese is for instance a typical isolating language, and Japanese a typical agglutinative one, we were able to measure the performance of our technique on radically different language types.

To be able to evaluate the technique with text from different domains, we also experimented with the first document collection of the Initiative for the Evaluation of XML retrieval [3], later referred to as the INEX IEEE collection [LT07], a 494Mb collection of 12,107 English-written computer science articles from IEEE journals. We only relied on the Keyword element of each topic, of which an example was shown earlier in Figure 3.1. Statistics about the collections are summarized in Table 3.2.

Generalities on (the lack of) language processing. An important point of this work is the development of language- and domain-independent techniques. This is put in practice in the following experiments. We have used no list of stopwords, and have applied no stemming. The only exception we made to this rule is in fact applicable to all languages: sentences are delimited by punctuation. We, hence, used every item in the text as a feature, with the exception of punctuation marks (e.g., periods, commas, parentheses, exclamation and question marks). For English, we dealt with sequences at the word level (space-delimited), whereas for Asian languages, we worked at the character level (to be precise, the few characters coded on more than one byte were represented by a concatenation of the corresponding hexadecimal codes). To illustrate this, a sample of a Japanese document is shown in Figure 3.2, and the corresponding hexadecimal representation is produced in Figure 3.3. This generalisation was performed on purpose, to underline the extent to which our approach is global, and does not rely on external knowledge.

We can verify the presence of full stops and observe that the two numbers occurring in the original document ("39" and "1997") are not replaced by hexadecimal

[2]NII Test Collection for IR systems, http://research.nii.ac.jp/~ntcadm/index-en.html
[3]INEX, `https://inex.mmci.uni-saarland.de/`

<DOC>
<DOCNO>JA-980101001</DOCNO>
<LANG>JA</LANG>
<SECTION>１面</SECTION>
<AE>無</AE>
<WORDS>742</WORDS>
<HEADLINE>［社告］「第３９回毎日芸術賞」決まる</HEADLINE>
<DATE>1998-01-01</DATE>
<TEXT>
　第３９回毎日芸術賞（１９９７年度）の受賞者が決まりました。この賞は当年度、優れた芸術活動をした個人・団体に贈るもので、各分野の多数の専門家のご意見を参考に毎日新聞社が選定しました。
……
</TEXT>
</DOC>

Figure 3.2: A sample Japanese document of the NTCIR collection.

```
a1ce bcd2 b9f0 a1cf a1d6 c2e8 39 b2f3 cbe8 c6fc b7dd
bdd1 bede a1d7 b7e8 a4de a4eb a1a1 c2e8 39 b2f3 cbe8
c6fc b7dd bdd1 bede a1ca 1997 c7af c5d9 a1cb a4ce bcf5
bede bcd4 a4ac b7e8 a4de a4ea a4de a4b7 a4bf . a4b3
a4ce bede a4cf c5f6 c7af c5d9 a1a2 cda5 a4ec a4bf b7dd
bdd1 b3e8 c6b0 a4f2 a4b7 a4bf b8c4 bfcd . c3c4 c2ce
a4cb c2a3 a4eb a4e2 a4ce a4c7 a1a2 b3c6 caac ccee a4ce
c2bf bff4 a4ce c0ec cce7 b2c8 a4ce a4b4 b0d5 b8ab a4f2
```

Figure 3.3: A sample of the representation of the Japanese document shown in Figure 3.2.

code and concatenated (on the second and third line of Figure 3.3). This is because they are formed of characters encoded on a single byte.

MFS extraction. Because we were dealing with multiple languages in an independent fashion, it was natural to rely on an equally language-independent algorithm for the extraction of sequences. We therefore applied $MFS\_MineSweep$ to all document collections using sentence subcollections formed with the k-means algorithm where k was uniformly set to 1 per $50,000$ sentences (see Section 2.1 for details).

3.1.4 Results and Discussion

The goal of our experiments were two-fold. First, we wanted to handle the numerous parameters induced by the technique and verify a number of assumptions on what those parameters shall be for different document genres and languages. We wanted to prove or disprove the assumptions that some language families shall benefit more from our technique than others; Agglutinative languages, where word order is less important, were notably expected to benefit from our approach. Also, we expected

better results with specialized documents versus non-specialized, due to more specific terminology.

The other question was of course, to find out whether our phrase-based similarity measure induced better retrieval performance than simply using bigrams, as is commonly done.

Impact of language. For English, the experiments permitted to confirm that no two English words are connected if there are more than 5 other words between them, as was shown by Jones and Sinclair [JS74]. Indeed, varying the maximal distance between words to any value higher than 5 had no impact on the results.

We were under the assumption that our technique would benefit most to languages where the relative positions of words are less important, that is, typically, agglutinative languages, where word-modifying morphemes are typically *agglutinated* to the corresponding word, meaning that changing its position seldom changes its role in the sentence. Example agglutinating languages are Turkish, Finnish, and Japanese. Respectively, for isolating languages, where relative word positions are most important, we did not expect great performance from our matching technique. This situation is that of isolating languages, such as Chinese, Korean, Samoan, or to a lesser extent, English.

The confirmation of our assumptions was clear for Chinese, whose isolating nature was shown by the best performance observed when only adjacent non-inverted pairs were considered. In turn, the agglutinative nature of the Korean language was shown by the domination of the runs in which few restrictions were applied on relative word positions. Surprisingly, however, allowing for the inversion of the word pairs affected the results negatively both for Korean and Japanese, in spite of the very typical agglutinative nature of the latter language. Therefore, we remained inconclusive with respect to the idea that the family of agglutinative languages should benefit more from our technique than the family of isolating languages.

Impact of genre. By similarly opposing the differences between the MAP results of word terms vector space model (WVSM) and of our technique for the specialized INEX collection and the NTCIR English news-feed collection, we could observe that only the INEX collection obtains better results with our technique (+4.2%), while with the English NTCIR collection, a MAP decrease was observed (-13.6%). This confirms the assumption that **our technique is more beneficial to specialized collections**.

Our Phrase-based similarity vs. Bigrams. To truly evaluate the impact of our technique and not the impact of MFSs as descriptors for document retrieval, we compared the results of our approach ($SEQ - New$) to those of an approach where the adjacent bigrams occurring in the set of phrasal descriptors are added as extra dimensions of the vector space model ($SEQ - Bigrams$). The comparison of our technique to SEQ-Bigrams shows a decrease for both English collections, -11.4% for the INEX collection and -18.0% for NTCIR-EN. A very clear improvement is, however, observed for all three Asian languages. For Japanese, the MAP improvement is as high as +51.2%. Comparably high benefits are observed for Chinese (+42.0%) and Korean (+42.3%).

The main difference between the way we processed the English and Asian document collections is that we formed words in the English collection, while we worked at the character level for the three Asian collections. This difference of granularity may be a good explanation for the clear improvement brought by our technique in one case, and for the harm it did in the other.

This indicated that the benefit of our approach is linked to the granularity of the items at hand, with same-sized sequences of small items being more useful than those of large items. In other words, a sequence of 5 characters is more beneficial than a sequence of 5 words, because a sequence of 5 words is too specific. Our technique hence permitted a higher improvement versus a 2-gram baseline, when the grams represent smaller items, e.g., characters rather than words.

3.1.5 Conclusion

We developed a novel technique for measuring the similarity of phrasal document descriptors and combining it to word-based vector space similarity measures. We applied our technique to the problem of document retrieval, where we compared the MFS-based phrasal representations of documents to sets of keyphrases describing user needs.

Due to a number of adjustable parameters, our method allows accounting for occurrences of the words of a phrase over a longer span, or in a different order. These usages may be gradually penalized, as compared to an exact phrase occurrence, i.e., adjacent words occurring in the same order. This approach permits taking a wide variation of word usages into account.

It notably deals with the problem of overlapping phrases, as described by Vechtomova [Vec05]. She states the problem of overlapping phrases as the fact that, given a query ABC, a document containing the exact match ABC and a document containing AB and BC separately both obtain the same score at the state of the art. A subsequent issue is that the weight of the word B becomes artificially heavier than that of A and C, because B is present in both pairs AB and BC. Our technique permits eradicating this problem, since it can also take the pair AC into account. Hence, the distance one between A and C in the first document (with ABC) ensures that it gets a better score than the second document (with AB and BC). Another consequence is that the weights of A and C are increased along with that of B, avoiding to unbalance the individual term weights within the phrase. A weakness, however, remains with this approach: the word terms that belong to a long phrase appear in numerous subpairs, and hence their artificial weight increase is more important than that of a word occurring in a shorter phrase. Notably, the weight of individual word terms that do not occur in a keyphrase is made lower in comparison to that of word terms occurring in a keyphrase. A solution would be to normalize the weight of terms upon the number and size of the phrases they occur in. This problem is not straightforward, as was suggested by work of Robertson et al. [RZT03] who proposed to subtract the individual weight of words that occurred redundantly in keyphrases and obtained very disappointing results.

As compared to throwing all descriptors in a bag of words, our similarity measure greatly improves the results for the NTCIR collections in Chinese, Japanese and Korean, with encouraging amelioration ranging between +42% and +51%. This

suggests that exploiting languages at a character level may well be the appropriate case for applying our technique with worthwhile improvement.

3.1.6 Perspectives

We present a project, aiming to study the use of phrases in commercial search engines, and looking to determine classes of keyphrases, with the potential to adjust phrase-based similarity parameters (of our algorithm or another) for optimal retrieval performance. This plan would constitute a relevant framework for doctoral research.

Adjusting parameters. As we have seen in Section 3.1.2.3, our matching technique functions with a number of parameters to be applied to the keyphrases, namely, inversion and non-adjacency penalties, duplication bonus, and maximal matching distance. We experimented with common-sense guesses of which parameters need to be emphasized for different types of documents and languages. This exploratory work gave us numerous interesting hints.

Originally, we planned to automatically detect whether a corpus is written in a rather agglutinative or isolating language, and adjust the parameters accordingly. The feasibility of such an automatic detection sounded realistic, assuming for instance that agglutinative language corpora would contain more distinct words, with other things equal. In that case, word inversion would be switched on and maximal distance increased. However, our exploratory work showed that things were not that simple.

The determination of parameters is nonetheless key to the performance of our approach, and the impact of their values needs to be explored in a systematic way. While machine learning can be a solution, the learning cost at querying time (when users expect a quasi-immediate answer) shall be a hindrance. What shall be done is learning from the corpus, at indexing time, which shall be adequate parameters, with the query as a source for potential adjustment.

This task is however very challenging, as we will see that even the question of when to use phrases and not is a long-lasting open question in information retrieval.

Predicting queries in which phrase-based similarity is beneficial. As we have seen, even given adequate and expert-validated MWUs, information retrieval systems seldom manage to obtain an improved performance when using phrases.

The exploitation of phrases in IR is astonishingly a very open problem despite decades of research. As underlined by Bruce Croft in a keynote talk [Cro05], it has been observed that the document retrieval performance is improved by the use of phrases for "some" queries, while for "some others" the impact is negative. A current trend in IR is to predict and adapt to difficult keyword queries [CSYT05, IS10, CYTDP06]. However, no such work has been developed towards predicting which queries will benefit from the use of MWUs and which ones will not.

The uncertainty as to which set of words should be treated as a keyphrase and which should not is the key reason why commercial search engines are only exploiting explicit keyphrases. Obviously, being able to "guess" the cases in which the use of keyphrases will benefit to IR applications would lead to a straightforward and

safe way to improve their performance. Current state of the art for commercial search engines relies on query data: if a sequence of words is repeated in a sufficient number of queries, it should be considered a keyphrase. Such a pure statistical view is algorithmically robust, but it shall evidently miss rare keyphrases queries.

In addition, following Croft [Cro05], due keyphrases are not necessary ones for which it is a mistake to handle components as distinct terms. In other words, that a query is a true multiword unit does not automatically imply that phrase-based similarity shall be beneficial.

Towards an IR-oriented classification of phrases. A first step shall be an exhaustive investigation into the recent field of query difficulty prediction, to work on the characterization of the different types of MWUs. Intuitively, it is clear that rigid word combinations (for example "Los Angeles") are more safely treated as keyphrases than more flexible collocations, such as "animal protection" which may be substituted by "protection of animals" or even "mammal protection". We want to classify the different types of MWUs from a language-independent perspective and analyse the performance improvement brought by the different types.

By this means, we hope to be able to detect classes of MWUs that can be safely considered as cohesive units for information retrieval applications. The ability to recognizes the MWUs that are mostly likely to induce a performance improvement would permit safely crossing the line of an application of MWUs research to real-world applications.

It is important to underline that the multilingual aspect of this project is a major challenge. There exist distinct MWU classifications for most different languages, but what we are willing to do is to statistically define language-independent categories, hence keeping this work within the efficiency constraints of commercial search engines.

Analysis of query logs. To lead this work, an interesting track is to study the use of Multiword Expressions in Web queries. We intend to explore the Web search click data and compile statistics about multiword queries. For instance, we wish to investigate the proportion of explicit multiword queries (quoted), and among those: how many are true multiword expressions, or contain multiword expressions? What level of satisfaction do these queries trigger (this "level of satisfaction" being inferred from reformulation and clickthrough data). Is there a relationship between satisfaction and true multiword expressions (are users more or less successful with such queries or others?)? Also, we would like to investigate reformulation patterns: how often do users reformulate an MWE query into a non-MWE query, how often do they reformulate a non-MWE query into an MWE query? How often do they add or remove quotes?

An ideal collection for this is the MSN Search query Log excerpt (RFP 2006 dataset), containing 15 million queries sampled over one month and containing queries, returning lists of answers and the corresponding links that were clicked through, all with time-stamps. Within the Workshop on Web Search Click Data 2009 [WSC09], together with Rosie Jones (then at Yahoo! Research in Sunnyvale, California), we gained access to this data set and designed the protocol of experiments, but unfortunately we lacked the time and resource to complete the

a. *"That there be freedom of movement established between Gaza and the West bank", she said.*

b. *"It is very important for ordinary Palestinians that there be freedom of movement established between Gaza and the West bank".*

c. *"That there be freedom of movement established between Gaza and the West bank", she added.*

d. *She said it is very important for ordinary Palestinians that there be freedom of movement established between Gaza and the West bank.*

Figure 3.4: A sample set of 4 paraphrases.

corresponding work.

In 2012, the workshop will be run again[4], this time with data from Yandex[5], the Russian search engine company.

3.2 Multiple Sequence Alignment of Text

An interesting application of Maximal Frequent Sequences was brought to light by collaboration with Gaël Dias, Rumen Moraliyski and João Cordeiro from the University of Beira Interior in Covhilã, Portugal.

This collaboration resulted in a publication in the Journal of Natural Language Engineering in 2010 entitled "Automatic Discovery of Word Semantic Relations using Paraphrase Alignment and Distributional Lexical Semantics Analysis" [3].

3.2.1 Motivation and Context

The main idea of this collaboration was to exploit paraphrases to detect words with related meanings, under the assumption that words occurring in similar contexts share a strong semantic link. This principle has long been ground for the use of text clustering techniques, notably in the field of word sense disambiguation (WSD) [MS99].

The application of this work experiments with this idea, under the assumption that pairs of sentences in which one word can be substituted by another are a strong indication that these words may share very close meanings.

Paraphrases are sentences sharing an essential idea while written in different ways. A corpus of paraphrases is therefore ideal for the detection of terms substitutions in similar contexts.

The work led in Covhilã permitted developing of a state of the art procedure for paraphrase extraction and clustering, using the *Sumo-Metric* [CDB07]. A sample set of extracted paraphrases is given in Figure 3.4.

[4]Workshop on Web Search Click Data 2012, `http://research.microsoft.com/en-us/um/people/nickcr/wscd2012/`

[5]Yandex, http://www.yandex.com/

Once a number of paraphrase sets have been extracted from a corpora, our goal was to learn TOEFL-like tests, i.e. clusters of words where there is a target word and a short list of semantically related candidates, predominantly in paradigmatic relations with the target. This can be done by aligning the paraphrases, and this is where I intervened with MFSs, introducing an MFS-based one-pass method for multiple sequence alignment.

3.2.2 State of the Art in Paraphrase Alignment

My goal was to align the paraphrases inside each cluster, detecting their common parts so as to evidence what differentiates them. Our approach considers sentences as word sequences and therefore reduces the resulting problem to that of multiple sequence alignment. In the field of bioinformatics, a sequence alignment is a way of arranging the sequences of DNA, RNA, or protein to identify regions of similarity that may be a consequence of functional, structural, or evolutionary relationships between the sequences. If two sequences in an alignment share a common ancestor, mismatches can be interpreted as point mutations, for instance. The alignment of sequences is performed to evidence their common and distinctive parts, possibly taking gaps into account [Not07].

Similarly, in the field of natural language processing, sequence alignment allows to observe variations in language use, and is particularly useful for similar text fragments, such as paraphrases [BL03]. But while there are several efficient techniques for multiple sequence alignment in the field of bioinformatics, they actually aim at slightly different problems. Indeed, biosequences to be aligned are typically few, very long and with limited vocabulary (e.g., there are only 20 amino acids, and only 4 molecules containing nitrogen present in the nucleic acids DNA and RNA, designated by the letters A, C, G, and T). In comparison, paraphrases are more numerous, shorter and with a larger vocabulary, since very few words are repeated within the same sentence. Consequently, most of the techniques to render the multiple sequence alignment more efficient are not appropriate for paraphrases [BL03].

3.2.3 Contribution

We therefore designed a 2-phase approach to efficiently align a set of paraphrases. In the first phase, we extract the Maximal Frequent Sequence set (MFS) of the paraphrases, that we later use as a pivot for the one-pass alignment of the paraphrases.

3.2.3.1 Maximal Frequent Sequences for Aligning Paraphrases

Maximal Frequent Sequences were extensively described in Section 2.1. A frequent sequence is defined as a non contiguous sequence of words that must occur in the same order more often than a given sentence-frequency threshold. MFSs are constructed by expanding a frequent sequence to the point where the frequency drops below the threshold.

It is worth to remind that this technique does not require any preprocessing. For instance, neither stemming nor stopword removal are necessary. This way, we can assign a set of MFSs to each set of paraphrases, without any specific preparation.

The ability to rely on a gap of variable length, an important specificity of *MineMFS*, is a very strong point in this application. It indeed allows to take into account language variations of different length, while state of the art techniques impose fixed distances, as we have seen in Section 2.1. For instance, In the following four French sentences:

a. *J'aime l'informatique.*

b. *J'aime beaucoup l'informatique.*

c. *J'aime énormément l'informatique.*

d. *J'aime vraiment beaucoup l'informatique.*

An MFS of frequency 4 "*J'aime l'informatique*" shall be extracted, efficiently gathering variations of length 0, 1 and 2 (ϵ, *vraiment, énormément, vraiment beaucoup*).

Now let us assume we are to align the 4 paraphrases in Figure 3.4. This set contains one MFS of frequency 4: "*freedom of movement established between Gaza and the West bank*". With this simple example, we get a reminder of the compact descriptive power of MFSs. Indeed, the 10-word sequence *"freedom of movement established between Gaza and the West bank"* would need to be replaced by $\binom{10}{2} = 45$ word pairs. With MFSs, no restriction is put on the maximal length of the phrases. Thus, we can obtain a very compact representation of the regularities of texts. So, by extracting the MFSs of a cluster of paraphrases, we obtain a compact sequential description of the corresponding paraphrases, i.e., a "skeleton" of the cluster that may be used for alignment.

3.2.3.2 Multiple Sequence Alignment with MFSs

Given the corresponding set of MFSs, we can extract the commons and specifics of a set of sentences, very efficiently, in one pass.

Thanks to the sequential property of MFS, we know that, passing in parallel through each of the paraphrases, we are bound to encounter the word "freedom", and that any word encountered before that is not common to all sentences. Once "freedom" was encountered, we know that we are bound to encounter the word "of", and that any word encountered before that is not common to all sentences. And so on, until the last word of the MFS. This way, we can easily and efficiently obtain the resulting alignment presented in Figure 3.5.

[{1,3:*that there be*}{2:*it is very important for ordinary Palestinians that there be*} {4:*she said it is very important for ordinary Palestinians that there be*}] freedom of movement established between Gaza and the West bank [{1:*she said*}{3:*she added*}]

Figure 3.5: The alignment corresponding to the sentences of Figure 3.4. Word sequences without brackets are common to both sentences. Others are specific to the sentences whose numbers precede the colon sign.

A final step was to form TOEFL-like test cases by gathering aligned segments with the same left and right MFS context. Further practicalities are not given here, since this manuscript is to focus on its author's achievements. Please refer to the aforementioned article for full details [3].

3.2.4 Conclusion

This collaboration permitted an innovative approach for word semantic relation extraction within a proposal differing from all other research presented until then as it tried to take the best of two different methodologies, i.e., semantic space models and information extraction models. As most of my work, it is language independent and can be applied to extract different semantic relations. It extracts relations between infrequent word senses. It limits the search space and it is completely unsupervised.

The MFS extraction and following one-pass alignment technique were especially adequate in such a language-independent framework. The main weakness of MFS extraction, its computational complexity at worst, could be overlooked since the sets of paraphrases contained small numbers of highly similar sentences, a special case that is "easy" on the $MineMFS$ algorithm.

The final results of the technique are detailed in the aforementioned paper [3]. In particular, as many as 35% of the constructed TOEFL like test cases contained close semantic relations. The methodology is also not hindered by low frequency words and discovered 88 synonymous word pairs not listed in WordNet. Compared to other methods that create long lists of words related in unspecified way, the methodology extracted very short lists of candidates in paradigmatic relation with the head. Those lists could easily be scrutinized by a human expert in computer aided thesaurus construction.

Finally, we applied a number of contextual similarity measures over the set of 372 tests. The fact that the preceding step yielded tests with few candidates allowed recall of 75% on detecting *Synonyms* and 58% on *Is-a* by the Global strategy over the Cosine-PMI combination, 71% on *Siblings* by the Product strategy over the Lin model [LZ03] and 49% on *Instance Of* by the Local strategy over the combination Cosine-TfIdf.

3.2.5 Perspectives

Maximal frequent sequences proved very relevant for this application. In particular the ability to extract sequences while taking into account a gap of variable length permitted taking into account language variations.

On the other hand, using word order as the only constraint naturally triggers the following subsequent weakness: the alignment fails in a number of cases when the paraphrasing effect is achieved through word order change. For example, the sentences

[{1:*The median price of an existing single family home dropped* 2.5 *percent from September 2005, the biggest year on year drop since record keeping began in 1969*}] the national association of realtors said [{1:*in Washington*} {2:*existing home sales*

declined for the sixth consecutive month in September while the median price fell 2.5% *year over year, the biggest decline on record*}]

are perfect paraphrases, but this is a common case when two long sentences are aligned around a single sequence that refers to the common agent.

Even when the alignment is anchored in many points there is still the option of conjunction rearrangements or even syntactic structure alterations such as the following alignment:

[{1:*He found*} {2:*This revealed*}] that [{2:*their*}] sperm [{1:*count, viability, motility*} {2:*declined steadily in number, quality*}] and [{1:*shape declined*} {2:*ability to swim*}] as mobile phone usage increased.

In order to avoid this kind of alignment and the consequent bad test cases, discourse analysis is necessary as well as reconstruction of the intended message by means of anaphora resolution.

A mixed approach shall be envisaged, where MFS of frequency n would be viewed as rigid, while subfrequent sequences may be experimentally permuted for comparative alignment.

3.3 Structured Information Retrieval

Documents can nearly always be split into coherent parts and subparts of different depth, with some of these elements possibly being interleaved. This hierarchical structure can be implicit, but thanks to the development and widespread use of markup languages, it is often explicit, simplifying the development and usage of modeling techniques that take document structure into account.

This section will present EXTIRP, the XML IR system developed under my lead [31]. We will further summarize the subsequent uses of the system, its extensions and results.

3.3.1 Related Work and Motivation

The exponential growth of the amount of textual information in electronic format required explicit ways to express the structure of documents. This is usually done through a markup language that follows strict rules. The use of explicit structures that follow strict grammars facilitates the exploitation of this new data by automatic approaches. We will now present a few such mark-up languages and the ways they were tentatively exploited in IR.

WWW and HTML documents. The World Wide Web (WWW) is an enormous source of structured information. The particularities of its formatting language, the *HyperText Markup Language* (HTML) [RLHJ99], have been an early and soon major center of interest for the web retrieval community. Thanks to HTML, web pages are highly structured. Different levels of headers can be marked-up, and numerous tags permit to identify important pieces of information. Another important strength of HTML is the possibility to include pointers between web pages, the so-called *hyperlinks*.

The markup of a text fragment offers a clear delimitation and extra knowledge about the meaning and importance of the text that it contains. Further, the markup elements in HTML are predefined and their number is fairly low. It is therefore simple to create rules for text encapsulated in any one of those tags. For document representation, the general technique of exploitation of the text markup of HTML webpages is to increase or decrease the weights of occurrences of words, depending on the tags they are encapsulated in. Cutler et al. [CSW97] proposed to use the structure of HTML documents to improve web retrieval. They studied the set of distinct HTML elements and assigned them into one of 6 disjoint classes. Each set of elements was associated to an importance factor that was reflected in the document model by modifying the weight of words, depending on the element class in which they occurred. In other words, term weights were linearly combined depending on the tags within which they occurred. Ever since, a number of techniques have been based on the same principle (see the latest TREC Web tracks [CCSC10, CCSV11] for a recent overview).

In addition to the markup encapsulating text, there is another important kind of information that can be taken into account in modeling a set of HTML-structured documents, that is, their hyperlink structure. Undeniably, if the author of a document X has decided to place a pointer towards the document Y, the content of X gives us information about the content of Y, and reciprocally. Further, in very large sets of documents, we are more inclined to *trust* documents that have many pointers towards them. The rationale behind this is that the authors of the pointers must have found these documents worthwhile, and hence they must be more important than others. This idea permitted major breakthroughs in web retrieval. The techniques implementing this idea relied on the same principle: representing the documents by a standard vector space model, *and* attaching an importance score to each document, based on hyperlinks (the more incoming hyperlinks, the more important the document), see e.g. [CH03, CH04]. A well-known variation of this principle is the PageRank algorithm [PBMW98], whose main particularity is to account for the importance of a document to calculate the importance of the others. In other words, it does not only base the importance of a document on the number of incoming hyperlinks, but also takes the importance of the documents of origin into account. A more recent trend has been to cross the boundary between document representation and hyperlink-based measure of importance. Therefore, researchers have tried to exploit the relations between content and link structures. The first way to do this is to use the document content to improve link analysis, as in [Hav03, Kle99], while another approach is to propagate content information through the link structure to increase the number of document descriptors [QLZ+05, SZ03].

Content-oriented XML documents. The eXtensible Markup Language (XML) [BPSM+04] is a generalized markup language that allows for a more varied structure than HTML. From a technical point of view, an important difference between HTML and XML is that the set of elements in HTML is fixed and predefined, whereas there exists no general set of predefined elements for a given XML document. In fact, the elements and the way they should be used need to be specified in a separate declaration, the *Document Type Declaration* (DTD), that describes what hierarchies of elements are allowed in corresponding documents. XML is called a *meta-language*,

```
<article>
 <fm>
  <hdr>
   <ti>
    IEEE TRANSACTIONS ON KNOWLEDGE AND DATA ENGINEERING
   </ti>
   <volno>Vol. 15</volno> , <issno>No. 4</issno>
   <mo>JULY/AUGUST</mo> <yr>2003</yr>
  </hdr>
  <atl>
    Topic-Sensitive PageRank: A Context-Sensitive...
  </atl>
  <au> <fnm>Taher H.</fnm> <snm>Haveliwala</snm> </au>
  <abs>
   <p><b>Abstract</b>
     The original PageRank algorithm for improving the...
   </p>
  </abs>
 </fm>
 <bdy>
  <sec><st>Introduction</st>
   <ip1>Various link-based ranking strategies have...</ip1>
   <p>The PageRank algorithm, introduced by Page et...</p>
  </sec>
  <sec><st>Topic-Sensitive Page Rank</st>
   <ss1>
     <st>3.1 ODP-Biasing</st>
     <ip1>The first step in our approach is to...</ip1>
   </ss1>
  </sec>
 ...
 </bdy>
</article>
```

Figure 3.6: A sample content-oriented XML document.

because each DTD can actually define a different language. HTML, on the other hand, is just one language. Another particularity of XML is that it is also used in the database community. As opposed to database-oriented XML documents, the main focus of interest of the information retrieval community is *content-oriented* XML documents, i.e., documents that consist essentially of textual data. An example of such a document, from the INEX[6] collection, is shown in Figure 3.6. This document gives explicit information about the publication details of a journal article and its content is structured with labels to mark the beginning and the end of paragraphs (<ip1>), sections (<sec>), and their titles (<st>). This document can be represented by the tree shown in Figure 3.7. The absolute XML Standard Path (XPath) expression towards an XML element is an incremental string indi-

[6]available at http://inex.is.informatik.uni-duisburg.de/2005/

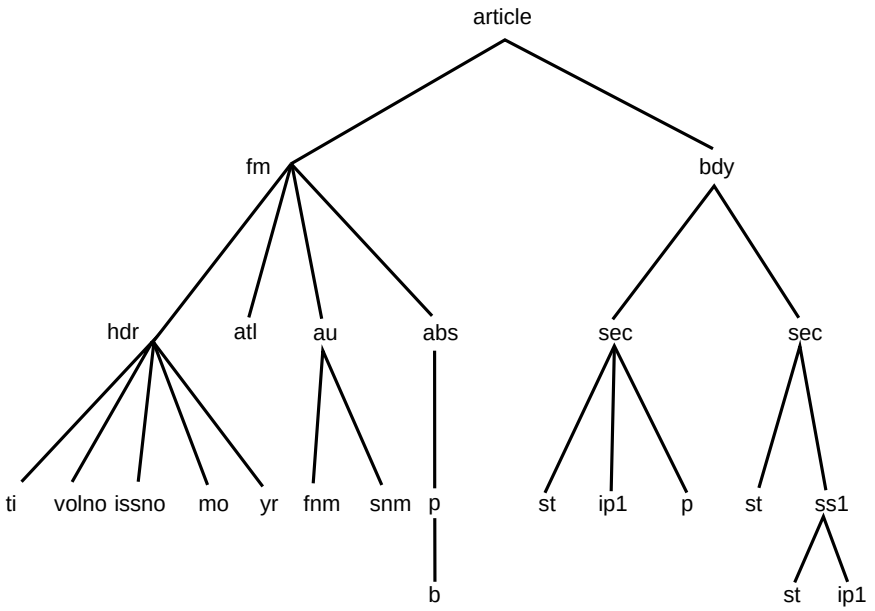

Figure 3.7: An example XML tree corresponding to the document of Figure 3.6.

cating the path to be followed to reach the element, starting from the root of the tree. Hence, the XPath expression of the element containing the word "Abstract" is "/article/fm/abs/p/b".

We will now give some insight in a few of the ways the structure of content-oriented XML documents has been used to improve the quality of their description.

Yi and Sundaresan [YS00] have used the structure of XML documents in a straightforward way. They concatenated to every word term its XPath of occurrence, and thus augmented the vector space model of another dimension for every distinct path of occurrence of a word term. They applied this document representation to the task of document classification and reported successful results. It must be pointed out that they experimented with a well-structured document set, where each element has a clear signification and high discriminative power. This is unfortunately not a typical situation.

Even though XML documents should ideally be provided with a DTD, real-life data often contains XML documents without one. Given a collection of XML documents without their DTDs, Nierman and Jagadish [NJ02] proposed a technique to evaluate the structural similarity between XML documents, with the aim to cluster together documents that originally derive from the same DTD. The measure of the pairwise similarity between two XML documents is based on their tree representations, and computed via a tree-edit distance mechanism. This technique performs well with respect to its goal. However, this approach focuses exclusively on the structure. From a content-oriented point of view, we naturally wish to integrate and combine content and structural information.

It is important to observe that XML, as opposed to HTML, was not designed as a language for formatting web pages. It has a much wider use, opening the door

to applications beyond this scope. XML is, for example, used in editing, where a document can be very long, for example being an entire book, or a collection of journal articles. This creates new challenges, as it is not always satisfying to treat full documents as independent entities. The clear delimitation inherent to the XML structure form a good background to deal with accurate subparts of the document rather than with entire documents only.

The aim of the INitiative for the Evaluation of XML retrieval (INEX) [FGKL02], started in 2002, is to address the problem of XML retrieval. Its initial case document collection was a set of 12,000 journal articles of the IEEE. The sample XML document given in Figure 3.6 is an article from the collection. This dataset is very relevant to the problem of XML retrieval. Suppose a user wants to find information about "datamining" in IEEE journals. A block of journal volumes certainly contains much relevant information, but is a too large answer to be satisfying. With this type of data, an information retrieval system needs the capacity to return only portions of documents. A simple way to be able to return XML document elements rather than full documents is to use the document structure, and represent each element by an independent vector. In that case, however, it is problematic to use standard weighting procedures, such as $tfidf$. The shortest elements will obtain the highest similarities with the user query, but to return a list of short italicized text fragments containing the word "datamining" will not satisfy the user's information needs either.

This clearly poses the problem of *granularity*. We should consider document fragments that are not too long but that are large enough to be able to stand alone as meaningful independent units.

Another new issue posed by XML retrieval is that of *content overlap* [KLdV04]. As we can observe in Figure 3.6, a paragraph may occur inside a section that may itself occur inside an article. The risk of directly representing XML elements as vectors is then to present the same highly relevant document portion several times, as it belongs to several overlapping elements. Clarke [Cla05] presented a technique to control the overlap. It consists in adjusting the score of lower-ranked elements, if they contain, or are contained, in higher-ranked elements. The rationale is to penalize elements that contain information that was already seen by the user, assuming she goes through answers in increasing-rank order, as is generally the case.

A full overview of the latest trends in XML retrieval is provided by the latest editions of the INEX workshop proceeding series [GKT09, GKT10, GKST11].

3.3.2 The EXTIRP System for XML Retrieval

In 2003, I was the leader of the EXTIRP[7] project, with the goal to address the granularity problem posed by (long) structured documents.

To address it, we defined the concept of Minimal Retrieval Units (MRU) [31], and tailored XML documents into trees, in which the leaves represent the smallest elements that can be returned (Section 3.3.2.1). The computation of the retrieval status value of the MRUs is described in Section 3.3.2.2. The representation of ancestors of the leaf elements is generated on-the-fly, by propagating dimension weights from children to parent elements. A weighting scheme is used to penalize

[7]EXacT coverage IR based on static Passage clusters, funded by the Academy of Finland.

the longest elements, so as to seek for a trade-off between relevance and exhaustivity, following the technique presented in Section 3.3.2.3.

3.3.2.1 Preparatory Procedures

Finding the most relevant text documents for each given topic is the basic problem to be solved in traditional IR. But, as the document collection is in XML format, we can identify two additional challenges that must be overcome before any traditional IR methods can be applied. First, the document collection consists of 125 XML documents, corresponding to journal volumes (containing each an average of about 100 scientific articles). These alone are too big to be retrieved on their own. Therefore, the collection is divided into smaller XML units which we shall call *XML fragments*. Second, the XML fragments contain all the XML markup that is present in the original XML format. Our goal is to convert the XML fragments into a text-only format where all XML markup has been removed without losing any of the information that is implicitly or explicitly coded in the XML structure of the original documents.

Division of the collection. The division of the collection was performed at two different levels of granularity called section-level and paragraph-level divisions. The levels for these two separate divisions were defined manually by looking into both the XML DTD and the XML documents. For example, the division into section-sized fragments concerned the following XML elements: `sec, fm, bm, dialog, vt`. In the document tree, all of these elements are close descendants of the `article` element, and none of them have text node children. In a similar fashion, the paragraph-sized elements taken into account in the paragraph-level division are `p, p1, p2, ip1, ip2, ip3, bq`. These elements have text node children, and also, most of the text content of the collection is covered by choosing these elements. A similar approach with a different set of element names was chosen by Ben-Aharon et al. [BACG+03].

By carefully defining the set of similar elements for each level, we intend to approximate an unsupervised division into fragments that is based on structural features only. Moreover, concentrating on structural similarity and discarding the information about element names will set us free from any particular document type or DTD. One might argue that contextual information is neglected by ignoring information specific to the document type. We believe, however, that identical content should be valued equally whether its parent element is called `sec` (section) or `bm` (back matter).

Intra-document links create connections between related XML elements. For example, footnotes are linked to the paragraphs that have a reference to the footnote element. Other examples include figure and table captions, biographical and bibliographical information, and other out-of-line content. Fragmentation of the collection separates linked elements unless both ends of the link belong to the same fragment. To avoid this, we have included some of the referred content that would increase the informational value of the fragment. Again, finding the intra-document links is possible without knowing the document type by a careful analysis of attribute values.

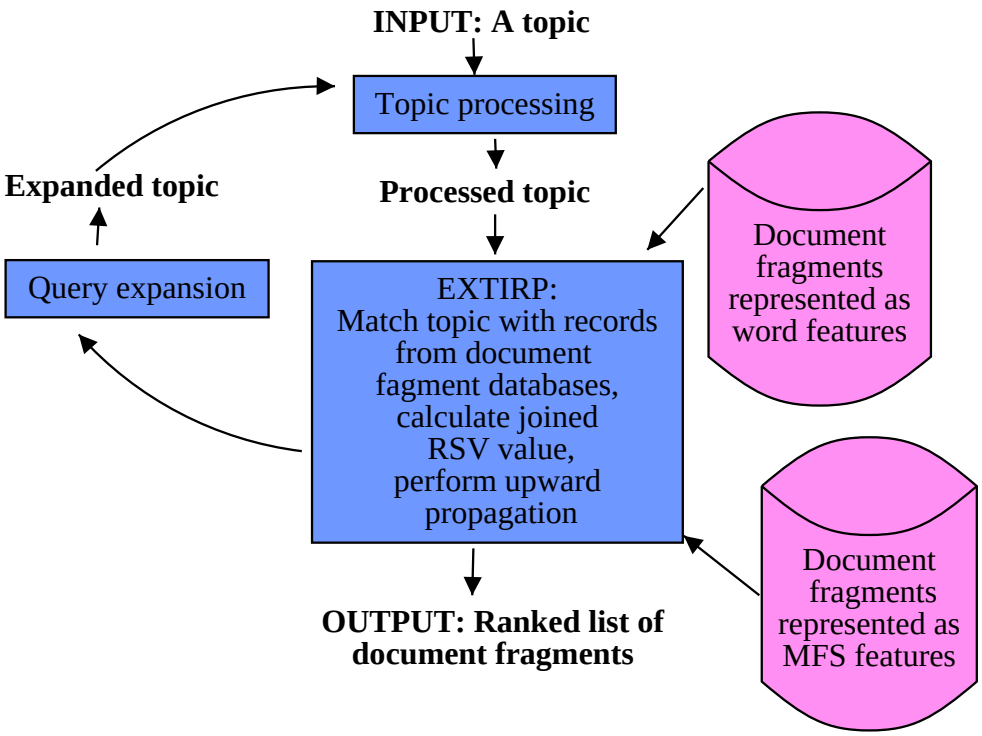

Figure 3.8: The system architecture for the scoring of MRUs. In this example, the MFS and Query Expansion modules are in use.

After the division, we have a collection of XML fragments. Each fragment is considered independent of the others, although information about the origin of the fragment is still included. The fragments can be combined later to make results with wider coverage, but dividing them further is hardly sensible as the size of a fragment is already supposed to be sufficiently small. In our system, these XML fragments constitute what are defined as *Minimal Retrieval Units* (MRU).

3.3.2.2 Variations in Scoring MRUs

The next task is to calculate the Retrieval Status Value (RSV) score of each MRU, a score which is to be propagated up the document tree.

There are many ways to calculate the RSV of a document with respect to a query. Throughout the years, we experimented with numerous approaches within the EXTIRP system. They will be covered in this section.

Figure 3.8 gives an overview of the process leading from raw text to the attribution of a score to each Minimal Retrieval Unit.

Basic Approach. Our initial baseline has been to rely on a straight-forward term-based vector space model with *tfidf* weights. In this case, the MRUs were represented by word features of the vector space model.

In 2008, we improved the system by updating our implementation of the word index with state of the art scoring functions for estimating the relevance of the MRUs [23]. We then replaced our implementation of the vector space model with the Okapi BM 25 implementation of the Lemur Toolkit [Lem03].

Query Expansion. In 2003, we experimented with query expansion (QE) for structured IR and integrated a QE module into EXTIRP [31]. The following is essentially the work of Lili Aunimo. While it is generally agreed that modern variants of query expansion improve the results of a query engine [BYRN99], there are many different ways in which QE can be performed. Some methods are based on relevance feedback, which can be blind or which can involve feedback from the user. In both cases, the QE approach is local because it is based on the retrieved set of documents. A global QE approach uses the the information derived from the whole document collection. Modern global QE methods usually use an automatically constructed thesaurus [QF93, CY92]. Other methods are based on manually crafted thesauri, such as WordNet, but experimental studies have shown that if the expansion terms from such theasuri are selected automatically, QE can even degrade the performance of the system [Voo94].

Our QE process can be considered a form of blind relevance feedback that has been inspired by the standard Rocchio way [Roc71] of calculating the modified query vectors. However, it is different from the traditional relevance feedback framework in that it takes into account only positive terms and no negative terms and in that it does not take into account all of the terms in the fragments, but only the best ones. This limits in practice the number of expansion terms per QE iteration between zero and ten. However, experimental studies have shown that even a few carefully selected QE terms can considerably improve the performance of a system [Voo92].

An outline of the process follows, whereas further details are available in the original paper [31]:

a. Run EXTIRP. The output from EXTIRP is a set of ranked lists of document fragments. There is one list per topic and the fragments are ranked according to their RSVs with regard to the topic.

b. Take the ten topmost items of each list.

c. Calculate the similarity threshold value.

d. For each topic do:

   (a) Take those fragments whose RSV is greater than the similarity threshold value. Make a list of words occurring in these fragments followed by their frequency count, and sort by frequency.

   (b) Take the ten topmost words and expand the topic with them.

   (c) Multiply the weights of the old terms by two and give the new terms a weight of 1.

e. Run EXTIRP with the expanded topics.

Maximal Frequent Sequences. We also implemented into EXTIRP our Phrasal RSV computation technique and its combinations, in the same fashion as presented in Section 3.1 which focused on the computation of inter-document phrase-based similarities.

The MRUs were then represented within two distinct models: 1) word features

a. *Initialisation*:

- $\forall n \in N$, score(n)=0
- $\forall m \in M$, score(m)=RSV(m)

b. *Iterate*:

- $\forall m \in M$: $\forall n_m \in N$ such that n_m is an ancestor of m, $score(n_m) = score(n_m) + score(m)$

c. *Final step*:

- $\forall n \in N$, score(n)=$\frac{score(n)}{(size(n))^{UPF}}$

Figure 3.9: Greedy upward propagation algorithm.

in the vector space model, and 2) maximal frequent sequences accounting for the sequential aspect of text.

When processing the queries, we computed two separate RSV values that were later combined following Lee [Lee95] (see Section 3.1.2.2): a Word RSV value based on the word index, and an MFS RSV value based on the phrase index.

The resulting combined RSV was then propagated to parent nodes following the procedure described hereafter.

3.3.2.3 Bottom-up Propagation

The result of the previous steps is the assignment of an RSV to each MRU of the document collection. In this section, we propose a technique for assigning an RSV to each of their ancestors.

Its principle is to propagate upwards the relevance value of each MRU, weighting it upon the size of the corresponding element. We define the size of an element to be the sum of the sizes of all its MRU descendants. In turn, the size of an MRU is the number of characters it contains.

Let A be an XML document, N the set of elements of A, and M the set of MRUs of A. We compute the score of each element $n \in A$ as shown in Figure 3.9. The UPF (*Upward Propagation Factor*) is a parameter that modulates the importance of the size of the elements. High UPF values give more penalty to big elements, and cause smaller ones to be promoted. On the other hand, if UPF=0, for any given article, the best score will always be given to the full article.

Because we assume that users go through answers in increasing rank order, we decided to avoid to propose them a document fragment they had already seen. Therefore, as a post-processing, we pruned every element having an ancestor with a higher rank. This implies for instance, that if UPF=0, the set of answers will only contain full articles.

3.3.3 Exploitation of Inline Elements

EXTIRP, developed in 2003 and actively used in 2008, also became the playground of Miro Lehtonen's Ph.D. thesis entitled "Indexing Heterogeneous XML for Full-

Text Search" and defended in 2006 [Leh06a]. The core of Miro's work focused on the exploitation of XML document structure for information retrieval. Unlike most of the work described earlier, however, the structure exploited lied at lower levels of the XML document tree.

Through the EXTIRP system, we implemented and experimented with a number of corresponding applications, two of which are most significant. The first one relied on the T/E ratio, described in Section 3.3.3.1, which was used as a filter to discard less interesting XML fragments from the selection of MRUs. The second application relied on the exploitation of inline elements to allow the extraction of maximal frequent sequences of better quality: more linguistically significant, and taking into account the emphasis stemming from the document mark-up, as detailed in Section 3.3.3.2.

3.3.3.1 Selection of XML fragments based on the T/E ratio

The selection of indexed fragments is based on two parameters: fragment size (min and max) and the proportion of Text and Element nodes (T/E measure). The algorithm starts from the document root and traverses the document tree in document order. The following steps are then iterated:

a. If the element is too big, move on to the next node and start over (from 1).

b. If the content looks like structured data (T/E<1.0), move on to the next node and start from 1.

c. If the element is too small, skip the subtree, move on to the next node and start from 1.

d. Index the element as an atomic unit, skip the subtree, move on to the next node and start from 1.

The resulting fragment collection does not cover the whole document collection. For example, parts of the documents that consist mostly of elements are discarded. Experiments on IEEE articles have shown that the algorithm works: it reduces the index size and improves retrieval precision [Leh06b]. When tested with the article collection, bibliographic and other data were successfully excluded from the full-text index.

Figure 3.10 shows the document with the lowest T/E value in the Wikipedia XML collection. The nested `cadre` elements are there either because of a faulty conversion from the Wiki format into XML or because of inconsistency in the source data. Because of the extra elements, the text content of this document was not included in the full-text index of EXTIRP, and thus it could not be retrieved, regardless of the query. However, the nested structures created with the proliferating XML elements are highly artificial.

This T/E based selection proved to be a good solution to handle this issue. Its usefulness is, however, strongly dependent on the type of structure. In [28], we argued that the good performance of the T/E approach with the Wikipedia was mostly due what we regarded as the main weakness of the collection: its structure

```
            </cadre>
           </cadre>
          </cadre>
         </cadre>
        </cadre>
       </cadre>
      </cadre>
     </cadre>
    </cadre>
   </cadre>
  </cadre>
 </cadre>
</cadre>
1987 "Recent Acquisitions", Tel Aviv Museum of Art
<cadre>
 <cadre>
  <cadre>
   <cadre>
    <cadre>
     <cadre>
      <cadre>
       <cadre>
        <cadre>
         <cadre>
          <cadre>
           <cadre>
            <cadre>
             <cadre>
              <cadre>
               <cadre>
                <cadre>
                 <cadre>
                  <cadre>
                   <cadre>
"Recent Acquisitions", Isreal Museum, Jerusalem, Isreal "20 Years of Occupation",
Bograshov Gallery, Tel Aviv, Israel (catelogue) "New Israeli Realism", traveling
exhibition, Omanut La'am (catelogue) "Artisrael, the 1980's", Leonard Perlson Gallery,
New YorkCity, "Haifa, Portrait of the City", Haifa Museum, Haifa Israel (catelogue)
"Fresh Paint", Tel Aviv Museum of Art (catelogue)
                   </cadre>
                  </cadre>
                 </cadre>
                </cadre>
               </cadre>
              </cadre>
             </cadre>
            </cadre>
           </cadre>
          </cadre>
         </cadre>
        </cadre>
       </cadre>
      </cadre>
     </cadre>
    </cadre>
   </cadre>
  </cadre>
 </cadre>
1989 International Biennale of Graphic Art, Ljublijana,
Yugoslavia (catelogue)
<cadre>
 <cadre>
  <cadre>
```

Figure 3.10: An excerpt from the document '3125748.xml' with a T/E value of 0.14.

...kernel trick has been applied to several algorithms in <link>machine learning</link> <link>machine learning</link> and <link>statistics</link> <link>statistics</link>, including...

Table 3.3: Two inline elements duplicated in place (XML attributes for link targets omitted).

for the INEX competition had been artificially built from the original structure stored by the Wikipedia foundation. This seemed to have been the main cause for the problematic cases that the T/E approach is solving. Such a safe guard is nonetheless very valuable, as experience shows that real-world data is filled with conversion problems, and the Web is for instance filled with HTML documents containing, e.g., unclosed tags.

3.3.3.2 Inline Element Duplication for Enhanced Phrase Extraction

Most of the XML markup in the Wikipedia articles describes either the presentation of the content or the hyperlink structure of the corpus, both of which show as mixed content with inline level XML elements. In these cases, the start and end tags of the inline level elements denote the start and the end of a word sequence that we call an *inline phrase*. These phrases include the anchor texts of hyperlinks as well as phrases with added emphasis, e.g., italicized passages. An exact definition for the XML structures that qualify was presented at the INEX 2007 workshop [26].

Intuitively, the inline phrases are highly similar to the multiword sequences that text mining algorithms extract from plain text documents. Therefore, the tags of the inline elements are strong markers of potential phrase boundaries. Because phrase extraction algorithms operate on word sequences without XML, we proposed to incorporate the explicit phrase marking tags into the word sequence by replicating the qualified inline phrases. This process, known as *in-place duplication* of the inline elements, adds phrase boundary indicators of an appropriate strength to the word sequence. Examples of such duplication are shown in Table 3.3.

MFS extraction and inline duplication. We observe that phrase repetition also repeats the proximity of its components. Thanks to its reliance on an unlimited gap, the MFS extraction algorithm is therefore more likely to extract co-occurring words originating from inline elements, following their artificial repetition.

Under the assumption that the inline elements contain true phrases, their repetition makes it relatively more likely that the MFS algorithm will in turn extract true phrases and their variations. The replication of the sequence "AB" into "$ABAB$" may further allow the extraction of "BA", "AA" and "BB", hence artificially augmenting the weight of the words A and B during the evaluation of phrase-based similarity.

Information retrieval experiments. We experimented with the INEX 2007 Wikipedia collection, which contains 107 topics. We ran the extraction of MFS following the in-place duplication of inline elements. We then ran our phrase-based similarity measure, and combined it with a word-based Okapi BM 25 implementation, as detailed in Section 3.1.

According to a topicwise comparison of the results, the best results were obtained when duplication affected both the phrase index and word index. Duplication further resulted in significantly higher Mean Average interpolated Precision (MAiP). On the INEX 2007 topics, duplicating the inline phrases led to a 10–15% improvement in MAiP. These experiments and results were presented in a short paper published at SIGIR in 2008 [14].

Further experiments hinted at slightly better improvement with triplication, but this time, the improvement was not statistically significant, hinting that duplication is probably sufficient to exploit inline markup [23].

3.3.4 Conclusion and Perspectives

We developed and implemented a generalised system for structured information retrieval. This technique exploits the logical structure of XML documents so as to give more focused answers to information retrieval queries.

The system, developed in 2003 under my lead in the Doremi group of the University of Helsinki, remained in use until 2009, year of publication of the latest EXTIRP-based paper [23]. During that period of time, it was used yearly as the experimental system for the participation of the University of Helsinki to the INEX XML retrieval initiative, and involved the development of numerous research experiments and corresponding software modules.

The EXTIRP framework is fully independent of the document structure. It notably does not require a DTD, which is a consequence of the flexibility of the definition of MRUs. It needs to be underlined that this independence from a strict schema is true to all modules and additions of EXTIRP, developed throughout the years: Query expansion, T/E-based filtering, inline element duplication, etc.

My doctoral work on the mining and use of multiword units was the synergistic reason to implement the phrase-based similarity measure (described in Section 3.1) within EXTIRP. Miro Lehtonen's idea of inline element duplication combined nicely with MFS extraction and phrase-based similarities, to allow a significant improvement in terms of retrieval performance.

Perspectives

Possibilities of subsequent future work are many in the field of structured information retrieval.

Duplication of inline elements in Web retrieval. As we have seen, we have been able to locate more high quality phrases in XML documents. Also, the correspondence between frequent multiword sequences and the high quality phrases of natural language was improved: we can extract fewer unnatural phrases composed of words that just happen to co-occur frequently. Because most (83.8%) of the duplicated content comes from the anchor texts of hyperlinks, we strongly believe that duplication when indexing phrases is also applicable to other hypertext documents such as HTML.

Increasing the weight of word terms contained in certain HTML elements is nothing new. This has long been done for HTML, mainly because it is a fixed language,

and because Web search is an enormous market. However, to the best of our knowledge, such a replication for the purpose of facilitating phrase extraction would be novel. Nonetheless, the computational cost of MFS extraction is a hindrance for a large scale Web application, and further research should rather aim at niche search.

Improvements of the basic EXTIRP model. There is a number of improvements to be achieved. One of them is to reuse the clusterings formed prior to the extraction of maximal frequent sequences, aiming at query optimization. The idea is that by comparing queries to centroids of MRU clusters, we will be able to efficiently skip large quantities of MRUs, without having to compute similarity measures for each minimal unit individually.

One concern in the EXTIRP process resides in the nature of the upward propagation formula, which is exponential in nature, meaning that a small variation in the UPF factor can cause a switch from a set of answers with a large majority of minimal units to a set of answers with a large majority of complete documents. Part of our future work is to explore the various ways to smooth this effect. However, doing so is difficult without new collections, as the current values have been functioning well for both the IEEE and Wikipedia collections.

Book Retrieval. It would notably be very interesting to run experiments with book retrieval, provided that a book collection be released to the academic world with sufficient structural mark-up. Current collections provide little more than physical structure (e.g., pages), whilst we would rather wish to exploit logical structure. This topic will be discussed at length in Chapter 4. I strongly believe that book collections are a key application for structured information retrieval and I am therefore eager to conduct subsequent retrieval experiments with such collections as soon as it becomes realistic. Plans for future research in the field of book retrieval are further discussed on page 116 of Section 4.5 .

3.4 Unsupervised Classification of Structured Documents

Document clustering has been applied to information retrieval for long. Most of this work followed the *cluster hypothesis*, which states that relevant documents tend to be highly similar to each other, and, subsequently, they tend to belong to the same clusters [JvR71]. Clustering was then applied as pseudo-relevance feedback in order to retrieve documents that were not good direct matches to the query, but that were very similar to the best results [Tom02]. In that case, documents are clustered before querying, so as to form document taxonomies.

The quantity of data organized with an XML structure keeps growing drastically. While XML document collections have essentially been data-centric, there have been more and more text-centric document collections. The necessity for tools to manage these collections has grown correspondingly. Clustering is one way to automatically organize very large collections into smaller homogeneous subsets.

In 2002, we experimented with a number of techniques for which no performance evaluation framework was available. The techniques were built on top of the vector

space model, which was enhanced with different types of textual and structural features. We proposed to combine textual and structural characteristics of documents at once and sequentially, in two successive steps.

We later presented the corresponding results in the context of the INEX 2006 document mining track and extended our contribution with the integration of a measure of the "textitude" of a structured document.

Because we require no document markup description (such as a document type definition — DTD), our techniques are particularly suited for experiments with several different collections, such as the ones used in the XML mining track: the IEEE journals collection and the Wikipedia collection.

We will first cover related work in Section 3.4.1, before presenting our contributions (Section 3.4.2). We will then discuss the concept of XML clustering and its evaluation in Section 3.4.3, and explain why the future prospects of this part of our work are fairly low on our priority list (Section 3.4.4).

3.4.1 State of the Art

Until the INEX mining track was launched in 2005 [DG07], most of the research on structured document processing was focused on data-centric XML (see for example [GM00] and [YS00]). One early motivation for XML document clustering was to gather documents that were structurally similar, so as to generate a common DTD for them. Nierman and Jagadish notably proposed a tree-edit distance as a structural similarity measure of XML documents [NJ02].

The birth of the INEX mining track in 2005 provided an experimental framework very much needed for the case of text-centric document collections [32]. This triggered research at the crossroads of information retrieval, machine learning and XML databases.

There are two main approaches to text-centric XML document clustering. One of them is to build models naturally close to the XML tree structure, such as neural networks [YHT+06], including self-organizing maps [KHT+07]. The other approach relies on a transformation of the document structure into a flat vector space representation, before applying well-known clustering techniques [32, DLTV05, VFLD06, CTT05]. Previous work has proposed to use element labels as the structural features, and to combine them into word term features in a common $tfidf$ framework [32]. Candilier et al. [CTT05] proposed more advanced structural features, such as parent-child or next-sibling relations. Vercoustre et al. [VFLD06] proposed to represent an XML tree by its different sub-paths, with features such as the path length, or the number of nodes it contains. An open problem for such techniques is to find a good way to combine the structural and textual features.

3.4.2 Combining Textual and Structural Features of Documents

The document model we used was the vector space model. We represented documents by N-dimensional vectors, where N is the number of document features in the collection.

a. *Initialization*:

- k points are chosen as initial centroids
- Assign each point to the closest centroid

b. *Iterate*:

- Compute the centroid of each cluster
- Assign each point to the closest centroid

c. *Stop condition*:

- As soon as the centroids are stable

Figure 3.11: Base k-means algorithm

Using this document model and the k-means algorithm, we performed our clustering experiments with various feature sets in one and two steps. We will now describe the clustering algorithm and then present the different ways we used it.

In other words, our contributions rely on the way we filled up the vector space and aggregated clusters, not in the document model used or the clustering technique proper.

3.4.2.1 Clustering technique: k-means

We chose to use the k-means algorithm for our experiments. *K-means* is a commonly used partitional clustering technique, where k is the number of desired clusters, either given as input, or determined in the loop. The algorithm relies on an initial partition of the collection that is repeatedly readjusted, until a stable solution is found.

In these experiments, we mainly decided to use k-means because of its linear time complexity and the simplicity of its algorithm.

Given k desired clusters, k-means techniques provide a one-level partitioning of the dataset in linear time ($O(n)$ or $O(n(log\ n))$ where n stands for the number of documents [Wil88]). The *base* algorithm presented in Figure 3.11 takes the number of desired clusters as input. In our experiments, we used the Cluto software package[8] to perform the k-means clusterings.

3.4.2.2 Document Collections of INEX

The INEX mining track, starting in 2005, provided two separate XML document collections. The first one is a collection of scientific journal articles from the IEEE Computer Society. The second one is a collection of English documents from Wikipedia [DG06]. Each collection further includes a set of categories C. Every document is assigned to a subset of categories of C that describe it best. The goal of the clustering task is to automatically produce a categorization that matches these

[8]CLUTO, http://www-users.cs.umn.edu/~karypis/cluto/

(*ideal*) assignments as closely as possible. We will now describe the two collections in further details.

IEEE. The IEEE collection has long been known as the "INEX collection", because it was the only collection in use in the main INEX track from the first INEX initiative in 2002 until 2006. It contains approximately 12,000 articles published in 18 different IEEE journals. They are mainly marked up with hierarchical and stylistic elements. The hierarchical markup typically indicates the beginning and end of sections, subsections and paragraphs, and possibly their titles, as well as figures and bibliographical references. Stylistic elements, for instance, are used to mark bolded text or mathematical formulas. A sample document was shown in Figure 3.6 page 64.

The categories that were used to partition the collection are the journals in which a document was originally published. Hence, every document is assigned to exactly one category.

As we pointed out already in 2002 [32], we believe that these categories are not fully satisfying, as the fact that a paper was published in a given journal does not necessarily mean that it is entirely irrelevant to every other journal. Among other things, such a strict interpretation means we should assume that a paper published in "Transactions on Computers" cannot possibly have anything in common with a paper published in "Transactions on Parallel&Distributed Systems".

Moreover, the IEEE collection contains documents of different types. The most common document type is scientific articles, but the collection also contains calls for papers, book reviews, keyword indices, etc. Regardless of their nature, documents published in the same journal are assigned to the same category. An intuitive issue with this "ideal" categorization is that any clustering assigning documents by their nature will be penalized in the evaluation process.

Wikipedia. The Wikipedia collection was new to INEX 2006 [FLMK07]. It contains 150,094 English documents from Wikipedia. The collection used in the mining track is a subset of the "main" Wikipedia collection as described by Denoyer and Gallinari [DG06]. The main collection contains 659,388 documents and covers a hierarchy of 113,483 categories. It contains about 5,000 different tags, with an average number of 161 XML nodes per document and an average element depth of 6.72.

The subset of the Wikipedia collection used in the INEX mining track consists of the 150,094 documents which correspond to one and only one semantic category. The categories were extracted from the Wikipedia portals, which include 72 semantic categories (the 113,483 categories mentioned earlier come from a different source, check [DG06] for details). After the removal of documents to which more than 1 category was attached, only 60 non-empty classes remained. This partition was used as the evaluation's gold-standard.

Naturally, we may express similar concerns as the ones we expressed earlier about the IEEE collection. The assumption that a document should belong to one and only one category does not seem right when we are handling text. To use a partition as our ideal classification implies the assumption that no two categories have anything in common. This can hardly be right when those categories are based on semantics.

3.4.2.3 The "Tags → Text" approach.

In 2002, while I was working on the use of structure for document access, we made the conjecture that "As structure is supplementary information to raw text, there must exist a way to use it, that improves the clustering quality" and decided to use the recently released INEX collection to experiment with various naïve approaches to represent text-centric XML documents for partitional clustering [32].

As mentioned earlier, the XML mining track started only in 2005 [DG07]. Therefore, while our work was integrated in INEX in 2002, it did not take place into a specific competition, but it rather defined a new problem, strongly related to that of XML retrieval.

Hence, as this was pioneering work, we started our experiments with "logical" simple baselines. As our goal was to take into account both the semantics of the text and its structural markup, we initially built two corresponding feature sets:

- **Text features only**: These features are the result of a typical (unstructured) text representation. The dimensions of the vector space are the single word terms.

- **Tag features only**: This representation uses the XML element labels as the dimensions of the vector space.

The other experiments were attempts to combine the information of unstructured text to that of the structural indicators. A simple way to do so is to combine the text and tag features into a single vector space: In other words, merging the bag of words and the bag of tag names. We named this representation "**Text+Tags**". This naïve approach served as a baseline combination of textual and structural data. Note that we prevented the confusion between word features and tag features (the "art" element name, referring to the mark-up of an article, cannot be confused with the word "art").

"Tags only", with a limited number of features, allowed for very fast clustering, but the results were weak. On the other hand "Text only" gave the best results, yet with a much larger number features (and hence, slower classification). The first experiments suggested that simply "throwing tags into the bag of words" (that is, augmenting the term-based vector space with tags) was inefficient. The increased number of dimensions in this "Text+Tags" approach slowed down the clustering process and gave mid-way performance, between that of the "Tags only" and "Text only" features.

We developed an alternative way to combine textual and structural features, through a 2-step approach, that we named "**Tags → Text**" (read "Tags then Text"). It permitted to obtain better results, by putting aside structural outliers before running the textual (semantic) classification [32]. The algorithm is described in Figure 3.12. To use tag features exclusively is very noisy when most of the XML elements have a purely stylistic role, as is the case in the IEEE collection. The "Tags → Text" technique permits to benefit from the structural information of documents, with the internal similarity threshold as a safe-guard. Only the most cohesive tag-based clusters are kept, while the rest of the clustering process is achieved based on text content.

a. *Input*:

- A document collection
- n, the final number of desired clusters
- σ, the internal similarity threshold

b. *Step one, tag-based clustering*:

- Based on tag-features only, perform k-means with $k = n$
- Keep the m clusters with an internal similarity higher than σ

c. *Step two, text-based clustering*:

- Based on text-features only, perform k-means with $k = n - m$

d. *Finally*:

- The m tags-based clusters and the $(n - m)$ text-based clusters are combined to form the final n-clustering

Figure 3.12: The 2-step approach: Tags → Text

In practice, this algorithm is as fast as a text-based n-clustering (often faster). This is due to the fact that tag-based clustering is very efficient thanks to a representation with a very small number of features.

Results. On the IEEE collection, we could observe, both in our initial experiments in 2002, and in the XML Mining track in 2006, that tag-based clustering is very fast, and that using the "Tags → Text" approach performs just as fast as using text features only, when not slightly faster.

For evaluation, the XML mining track defined 18 classes corresponding to the 18 journals where articles of the IEEE collection were published. In our original setting in 2002, we had set aside articles marked-up *volume* in a 19^{th} class, since they were not containing scientific articles but rather book reviews, call for papers, etc. We also led our evaluation without any assumptions about the number of clusters, and relied on internal measures such as purity and entropy of the clusters. The INEX mining track measures were assuming a fixed number of clusters, equal to the number of categories, and relied on precision and recall measures (hence strictly implying that the clusters be disjoint, which is very arguable when it comes to text classification).

In all experiments, "Tags → Text" outperforms "Text+Tags". This result is especially satisfying because both approaches use exactly the same features. Consequently, we got confirmation that the "Tags → Text" technique is a better way to integrate structural features into the clustering process. The explanation is fairly simple. The structural clustering can detect and put aside what we may call "structural outliers". Typically, in the IEEE collection, they are tables of contents and keyword indices of journal issues as well as calls for papers. In the Wikipedia collection, the outliers include lists (lists of counties by area, list of English cricket clubs,

etc.). However, to count on a small number of element names as unique document descriptors is obviously risky. This is why we ensure that only the most cohesive tag-based clusters are kept, by using a high internal similarity threshold.

However, while in our 19 classes setting, the fact that "Tags → Text" allowed to set aside structural outliers (such as call for papers, book reviews, ...) allowed it to obtain the top results, in 2006, with 18 journal classes, the "Text only" approach outperformed it. As we argue in the perspectives, the INEX mining track may be criticized for making the evaluation of the clustering of XML documents based on strictly textual categories. However, there is no evident reason to claim that this is wrong. We came to the conclusion that it all goes down to applications, a forgotten prerequisite for evaluation, which was totally left aside. The conclusion of our paper in 2006 (actually already hinted in 2002), was that future work would remain hypothetical without clear use-cases for the clustering of XML documents.

With the Wikipedia collection, introduced in 2005, and therefore new to our system, the conclusions were identical: "Tags → Text" outperformed "Text+Tags", and was faster, but the very best results were obtained with "Text only" features, which we found natural given the evaluation settings.

The overall results of the competition were very flattering for us, since, out of 5 participating institutions, we ranked 2^{nd} with the IEEE collection and 1^{st} with the Wikipedia collection. The only better-performing method on the IEEE collection was based on contextual self-organizing maps [KHT+07], and relied on supervised learning, which is arguable in a competition on unsupervised classification. Its performance is fairly close to our own best but its complexity makes it hardly scalable (the supervised learning actually needs to be restricted to a subpart of the IEEE document trees: the content of the "fm" element). The complexity of the technique is the reason why it did not participate in the experiments on the Wikipedia collection.

As compared to other competitors, one strong point in our approach is indeed that it does not use anything but the documents themselves. From a computational point of view, clustering is the costliest operation with a linear time complexity of $O(n)$ or $O(n \log n)$. Hence we had no problems shifting from one collection to the other and we do not expect difficulties in applying this work to new collections, whatever their structure and size is, since our techniques scale well.

3.4.2.4 Integrating a new structural indicator: The T/E measure.

Within our 2006 participation to the INEX mining track [27], in addition to revisiting our 2002 experiments [32] in a formal framework, we also introduced a new structural indicator in the context of the unsupervised classification process: the T/E measure [Leh06b].

The T/E measure. The T/E measure is a structural indicator of the proportion of "mixed content" in an XML fragment. In previous research, it has given us the Full-Text Likelihood of an XML element, based on which, the element could be excluded from a full-text index (remember Section 3.3.3.1). Although the values of the T/E measure are in the continuous range from 0 to ∞, the interpretation has come with a projection into a binary value space, where values greater than 1.0 provide evidence of full-text content. When treated as a feature for clustering XML

documents, the projection is unnecessary. Therefore, the whole value space of the T/E measure is available. The T/E measure is a quite reliable indicator when the XML fragment is relatively small, e.g. a paragraph of text or a small section. As the size or the heterogeneity of the fragment increases, a single T/E value starts to shift from being an exact indicator towards being an approximation. As an illustration, Figure 3.10 page 72 showed an example of a Wikipedia fragment with low T/E value.

Integrating the T/E measure into the vector space model. We integrate the T/E measure as an additional dimension of the vector space. Because our weighting scheme is based on inverted document frequency, the inclusion of the T/E value for every document would have a null effect. Therefore, we only gave a non-null value to the T/E dimension if the T/E measure was greater than 1.

This process led to another technique, named "Text + T/E". It allowed for a tiny improvement over the "Text only" approach with the IEEE collection, while it had a negative impact with the Wikipedia collection. Overall, the impact of the T/E value was statistically insignificant. A very natural and simple step in future work would be to increase the weight of the T/E dimension. However, as this would not lift our concern about the evaluation procedure, we did not proceed with further experiments.

3.4.3 Some Thoughts on XML Clustering and its Evaluation

The evaluation of clustering consists of comparing automatic unsupervised classification instances to a given "gold-standard". Finding such an ideal classification is very difficult, as there may be many ways to split a document collection that are equally valid and arguable. However, we believe that the gold-standards used in the evaluation of the INEX mining track are heavily oriented towards the textual content of the document, and far less towards their structural content. Therefore, it came as no surprise that our best results at the INEX mining track 2006 were obtained by using textual features exclusively.

3.4.3.1 Evaluation of clustering

There are two ways to evaluate clustering experiments. The first one is to use *internal quality measures*, such as entropy, purity, or cohesiveness. For instance, the cohesiveness of a cluster is the average similarity between each two documents in the cluster. The problem of these measures is that the computation of document similarities is strongly dependent on the document model. Internal quality measures are useful to compare clustering techniques based upon the same document model, but they are meaningless in most other cases. In particular, they cannot help as we wish to compare techniques based on the same algorithm but different feature sets.

In such situations, we must rely on *external quality measures*, such as recall, precision, or F1-measure. The latter were the official evaluation metrics for the clustering task of the INEX mining track in 2006.

3.4.3.2 Gold-standard

External measures are meant to compare every submitted clustering to a "gold-standard" classification. The more similar a run is to that standard, the better. Defining such a gold-standard is a great challenge.

We are not convinced that the gold-standard classifications that were used for the evaluation of the INEX mining track are optimal. The consequence of this is very important, because to improve a system's performance with respect to the F1-measure means to produce a classification closer to the gold-standard. If the gold-standard is weak, improving the performance of a system might actually require that a number of reasonable assumptions be compromised.

3.4.3.3 What is a good clustering?

One issue with the current ground truth is the use of disjoint clusters. This is an excessive simplification when we are dealing with text and thematic classes.

In fact, having to deal with thematic classes can also be seen as a problem. Since the motivation of XML clustering is to take structural information into account, we should also consider categories that are not solely based on semantics.

An empirical analysis of our clusters show that the technique "Tags $\rightarrow$ Text" manages to put aside outliers, such as tables of contents or call for papers in the IEEE collection. Our technique stores these into clusters of their own and performs text-based clustering with the remaining documents. We do believe that this is a good result for most uses of the document collection. Thinking of information retrieval, it is likely that a user performing a search on a scientific journal is looking for articles (or fragments thereof) rather than keyword indices or calls for papers. However, with respect to the current evaluation metrics, the effectiveness of a system taking this fact into account is weakened, because the gold-standards were built the opposite way: each call for papers belongs to the journal in which it was published. Hence, they are spread out uniformly in the ideal classification, while we actually built a system that puts all of them aside, in a category of their own.

The problem is that if the gold-standards were solely based on the document structure, separating calls for papers, tables of contents and regular articles, one would as well be able to argue that it does not make sense to have to categorize the call for papers for a data mining conference is in a different class from that of a data mining article. The key issue is there. Given a document collection, there are numerous "perfect" ways to classify the documents. The classification needs to be related to a certain need, but it may still be totally inappropriate in other situations.

This leads us to the conclusion that the intended use of XML document clustering needs to play a larger role in its evaluation, and hence a prior question needs to be answered: why do we do it? If the goal of XML clustering is to build a semantic-based disjoint taxonomy, then the current gold-standards are suitable. In order to detect DTDs automatically, we would need a structure-oriented gold standard. For information retrieval, we might use the per topic relevance judgements as the classes.

3.4.4 Conclusion and Perspectives

We developed a technique to cluster structured documents combining textual and structural features of documents. The combination process, in two successive phases, permits detecting outliers very efficiently, before performing text-based semantic classification. It is faster and much better-performing than the baseline approach and remains one of the main techniques for XML clustering to date, with a citation count that surprises its own authors [9].

The generality and scalability of our approach was underlined by the fact that we made no difference in the way we handled two radically different document collections, whereas many participants have been discouraged by the size and depth of the Wikipedia document collection (perhaps also by the lack of a DTD).

One weakness of our techniques is their flat use of the structural information. We created a "bag of structure" and implemented advanced ways to use it as a complement of the "bag of words", but we ignored the tree structure of the elements and did not either connect the words to their path in the XML tree. This is left for future work.

For both collections proposed at the INEX mining track 2006, we had the satisfaction to see our runs in the top ranks. Looking at the top 5 results for the two collections, the only run that was not ours occupied the 1^{st} rank for the IEEE collection. The "Tags $\rightarrow$ Text" approach was demonstrated to be more efficient at combining semantics and structure, than a baseline merger of the features, in spite of a tendency to contradict some of the arguable implications of the current evaluation system. Hence, in a more appropriate evaluation setting, we expect to observe the same phenomenon with an even greater margin. Evidently, this remains to be verified.

A source of concern should be the fact that our best performing runs were the ones that actually ignored the structural information. However, we feel that this only reflects the bias of the evaluation system. Indeed, micro- and macro-average F1 are measuring the closeness of a run to a theoretically ideal classification. However, the current "ideal" classifications in use are disjoint and thematic. Since there is no evidence that the classifications used as gold-standards are related to the structure of the documents, it is natural that the best performing approaches are the ones that simply ignore that structure.

An important point is that several classifications of the same collection may be perfect, depending on the context. We hence plead for placing the applications of XML clustering in the center of the evaluation process. This may be done by creating an ideal classification for every corresponding application, and/or by evaluating XML clustering indirectly, by measuring how much we can benefit from it in another task.

This is how we chose to put this line of work aside, as we felt that research on this problem was not addressed properly, and totally left user case studies out of play. Since we did not come up with clear applications ourselves, we focused on other research problems.

[9] 65 citations according to Google Scholar (checked on 22 February 2012).

3.5 Related Publications

Multiple Sequence Alignment. The procedure of MFS-based multiple sequence alignment is detailed in the corresponding part of an article published in 2010 in the international Journal of Natural Language Engineering (JNLE) [3]. The article itself describes the whole procedure, from paraphrase detection from a news corpus to the extraction and evaluation of semantic relations.

Computation of phrase-based similarity. The core of the technique is described in my Ph.D. thesis [36, 37] and related contemporary papers in the JADT conference [17] and in the multiword expressions workshop of ACL 2004 [30]. A mature version of this work was presented in an article published in the international Journal of Language Resources and Evaluation (JLRE) in 2010 [4].

As we have seen in this chapter, through the EXTIRP system, phrase-based similarity has been employed in a number of other publications that I have co-authored [14, 23, 24, 26, 28, 31]. Their content is overviewed in the next paragraph.

Structured Information Retrieval. The back-bone of the EXTIRP system is extensively described in the first related publication in the INEX 2003 workshop [31]. The generality of the approach and its notable independence from any kind of document schema was demonstrated when switching from the IEEE collection to the Wikipedia collection in 2006 [26, 28].

The beneficial results of the replication of inline elements were demonstrated in a short paper in SIGIR 2008 [14] and in a longer paper published in INEX 2008 [23]. We pleaded for the potential of structured document collections for the extraction of multiword units in a SIGIR workshop paper [24], which, as of INEX 2009, lead to a clear increase in the number of topics provided with an explicit keyphrase markup.

EXTIRP was also the weapon of choice for a number of personal papers by Miro Lehtonen [Leh05, Leh06a, Leh06b, Leh06c, Leh07], including his Ph.D. thesis [Leh06a].

Unsupervised classification of structured documents. Our original technique was described in the very first INEX workshop in 2002 [34]. Interrogations about applications, and the fact that my doctoral work started to clearly focus on sequential data, made us leave this line of work aside. However, the birth of the INEX mining track led us to revisit our method in 2006 and experiment with the T/E technique, which had proved beneficial for structured retrieval [27].

Chapter 4

Evaluating the Performance of Systems

At first glance, this part of the manuscript may seem like an outlier in the sense that it does not deal with the details of particular methods, but rather with methodological aspects of evaluation.

While the evaluation of extracted knowledge mostly resides in indirect evaluation (through applications), the process of evaluating the applications in itself is not always straightforward.

We will hereby describe the work that we conducted in the evaluation of book information retrieval and book structure extraction. Section 4.1 describes the context and motivation of the work on books, in the context of INEX, while the following two sections describe declinations of my work on evaluation.

More specifically, Section 4.2 describes works on the evaluation of Book Retrieval. This ongoing work started in 2007 and has been led by Gabriella Kazai from Microsoft Research Cambrige UK. The growing extent of this work means we were glad to welcome further research companions throughout the years: Monica Landoni (University of Lugano, Switzerland) since 2008, Marjin Koolen since 2009 and Jaap Kamps since 2010 (both from the University Amsterdam, Netherlands). Since 2011, the Book Search task is the main track of INEX, which will be colocated with the CLEF conference[1] starting from 2012.

Section 4.3 deals with works on the evaluation of structure extraction from digitized books which I led since it started in 2008. Co-workers in this project were Gabriella Kazai and, until 2010, the Document Layout team from Microsoft Development Center Serbia (Bodin Dresevic, Aleksandar Uzelac, Bogdan Radakovic and Nikola Todic). In 2011, Jean-Luc Meunier (Xerox Research Centre Europe) joined the organizing committee of the Structure Extraction competition.

After a summary of related personal publications in Section 4.4, we will finally present the conclusions and perspectives of this chapter on evaluation in Section 4.5.

4.1 Providing Digitized Book Collections

Libraries around the world and commercial companies like Amazon, Google and Microsoft are digitizing thousands upon thousands of books in an effort to enable online

[1] Conference and Labs of the Evaluation Forum, `http://www.clef-initiative.eu/`

access to these collections. The Open Content Alliance (OCA)[2], a library initiative formed after Google announced its library book digitization project, has brought digitization projects into the public eye, even though libraries have been driving digitization efforts for decades before that. Unlike library digitization projects, which are centered around preservation and involve the careful and individual selection of materials to be digitized, mass-digitization efforts aim at the conversion of materials on an industrial scale with minimum human intervention [Coy06].

The motivation for commercial companies' involvement is driven by their respective business models. For example, Amazon aims to increase its book sales by digitizing books and offering a 'search inside' feature. Google and Microsoft follow an advertising revenue model and provide search services over digitized books in order to generate traffic on their sites. This has led to the predicament where industry has taken a leading role, while academic research has been lagging behind.

The increasing availability of the full-text of digitized books on the Web and in digital libraries, both enables and prompts research into techniques that facilitate storage, access, presentation and use of the digitized content. Indeed, the unprecedented scale of the digitization efforts, the unique characteristics of the digitized material as well as the unexplored possibilities of user interactions make full-text book search an exciting area of research.

Motivated by the need to foster research in areas relating to large digital book repositories, see e.g., [KKMFW08], we launched the Book Track in 2007 as part of the Initiative for the Evaluation of XML retrieval (INEX)[3]. INEX was chosen as a suitable forum as searching for information in a collection of books can be seen as one of the natural application areas of focused retrieval approaches [KGT08], which have been investigated at INEX since 2002 [GKT09, GKT10]. In particular, focused retrieval over books presents a clear benefit to users, enabling them to gain direct access to parts of books (of potentially hundreds of pages in length) that are relevant to their information need.

4.1.1 State of the Art

Naturally, prior to 2007, there was little hands-on academic research in information access applied to digitized book collections, due to the lack of available corpora. An attempt to distribute such data from Google was announced in 2007 [Vin07] but the plan to distribute a collection of digitized books to the research community never materialised.

In 2008, a wide consortium gathered to start the 4-year IMPACT project[4], funded under the Seventh Framework Programme of the European Commission (FP7). The project gathers 26 national and regional libraries, research institutions and commercial suppliers. Most of the academic partners are from the field of library science.

The project is strongly centered on OCR problems, and does not actually deal with end-user applications. The main overlap with our work resides in the problem of structure extraction, which is the focus of the "Enhancement & Enrichment subproject". Its goal is "to make the OCR results more accurate and more accessible

[2]Open Content Alliance, `http://www.opencontentalliance.org/`

[3]`http://www.inex.cs.otago.ac.nz/`

[4] `http://www.impact-project.eu/`, visited 24 October 2011

(...) work on collaborative correction, descriptions of physical and logical structure (...)".

This particular part of the project work is obviously close to our work focused on the evaluation of structure extraction, and the connection will be further discussed in the corresponding part of this chapter (Section 4.3).

However, the main problem is that while this project provides tools to "improve access to historical text", it does not provide (and does not aim to provide) a book collection, since it relies on copyrighted collections.

The Book Track, which we initiated in 2007 is therefore the first book collection to be built and distributed freely for research purposes. The collection is described in the next section. We will later describe the challenges we have since set up and supported to foster research on this promising material.

4.1.2 Collection Description

The corpus of the INEX book track contains 50,239 digitized, out-of-copyright books, provided by Microsoft Live Search and the Internet Archive [22].

The collection consists of books of different genre, including history books, biographies, literary studies, religious texts and teachings, reference works, encyclopedias, essays, proceedings, novels, and poetry.

Each book is available in three different formats: portable document format (PDF), DjVu XML containing the OCR text and basic structure markup as illustrated in Figure 4.1, and BookML, containing a more elaborate structure constructed from the OCR and illustrated in Figure 4.2

DjVu format. An <OBJECT> element corresponds to a page in a digitized book. A page counter, corresponding to the physical page number, is embedded in the @value attribute of the <PARAM> element, which has the @name="PAGE" attribute. The logical page numbers (as printed inside the book) can be found (not always) in the header or the footer part of a page. Note, however, that headers/footers are not explicitly recognized in the OCR, i.e., the first paragraph on a page may be a header and the last one or more paragraphs may be part of a footer. Depending on the book, headers may include chapter/section titles and logical page numbers (although due to OCR error, the page number is not always present).

Inside a page, each paragraph is marked up. It should be noted that an actual paragraph that starts on one page and ends on the next is marked up as two separate paragraphs within two page elements. Each paragraph element consists of line elements, within which each word is marked up separately. Coordinates that correspond to the four points of a rectangle surrounding a word are given as attributes of word elements.

BookML format. The OCR content of the books has further been converted from the original DjVu format to an XML format, referred to as BookML, developed by the Document Layout Team of Microsoft Development Center Serbia. BookML provides richer structure information, including markup for table of contents entries. Most books also have an associated metadata file (*.mrc), which contains publication

```
<DjVuXML>
 <BODY>
  <OBJECT data="file..." [...]>
   <PARAM name="PAGE" value="[...]">
   [...]
   <REGION>
    <PARAGRAPH>
     <LINE>
      <WORD coords="[...]"> Moby </WORD>
      <WORD coords="[...]"> Dick </WORD>
      <WORD coords="[...]"> Herman </WORD>
      <WORD coords="[...]"> Melville </WORD>
      [...]
     </LINE>
     [...]
    </PARAGRAPH>
   </REGION>
   [...]
  </OBJECT>
  [...]
 </BODY>
</DjVuXML>
```

Figure 4.1: A sample DjVu XML document

(author, title, etc.) and classification information in MAchine-Readable Cataloging (MARC) record format.

The basic XML structure of a typical book in BookML (ocrml.xml) is a sequence of pages containing nested structures of regions, sections, lines, and words:

BookML provides a rich set of labels indicating structure information and additional marker elements for more complex texts, such as a table of contents. For example, the label attribute of a section indicates the type of semantic unit that the text contained in the section is likely to be a part of, e.g., a table of contents (SEC_TOC), a header (SEC_HEADER), a footer (SEC_FOOTER), or the body of the page (SEC_BODY).

The original corpus totals 400GB, while the reduced version is only 50GB (and 13GB compressed). The reduced version was created by removing the word tags and their attributes (coordinates, etc.) and propagating the values of the val attributes as content into the parent line elements.

4.1.3 Research Questions

Naturally, the creation and distribution of such a unique collection of digitized books allows to investigate numerous research questions, in all fields concerned with text and libraries.

In this subsection, we will try to summarize a sample of research questions that can be investigated thanks to the collection. In our opinion, these research questions can be grouped in 3 categories: 1) questions that did not exist before digital book

```
<document>
 <page pageNumber=‘‘I-N’’ label=‘‘PT_CHAPTER’’ [...]>
  <region regionType=‘‘text’’ [...]>
   <section label=SEC_BODY’’ [...]>
    <line [...]>
     <word val=‘‘Moby’’ [...]/>
     <word val=‘‘Dick’’ [...]/>
    </line>
    <line [...]>
     <word val=‘‘Herman’’ [...]/>
     <word val=‘‘Melville’’ [...]/>
    </line>     [...]
   </section>   [...]
  </region>     [...]
 </page>        [...]
</docment>
```

Figure 4.2: A sample BookML document

collections, 2) questions for which such collections allow for new approaches, and 3) questions that need to be asked again: do common assumptions still hold for book collections?

4.1.3.1 New questions

Several research questions sprung up only because of the emergence of large digital book collections. Such questions did not exist, did not matter, or could not be answered before. Many of them relate to the field of Human Computer Interaction (HCI), as their aim is to create further benefit and confort for users of digital books versus paper books.

Numerous applications are possible with digital books, that are not available with paper books: sharing annotations, searching, recommending to friends, defining words in context, using internal and external hyperlinks and other artefacts to facilitate navigation,...

The fact that books are digital and require a reading device also allows to monitor the way books are read. This further facilitates research in several fields of behavioural science.

An interesting application may be the summarization of book parts on the fly. A user may for instance want to skip a chapter of a book but still get its quintessence before reading the next one. Or the user may have read the first part of a book and want to continue reading it several months later. Digital books combined with adequate summarization techniques may then allow for a tailored synopsis to be built.

Digital books have one key advantage for document classification: They do not need to stand on shelves. Hence all the classifications of libraries become outdated, and the way to organize books, originally based on the idea of shelves, may now entirely be reconsidered. It is possible to sort the books by authors names AND by genre, and it is possible to let the same book on bioinformatics stand at the same

time on the (virtual) shelves of the biology and computer science sections of the libray.

For even the most simple applications, it remains unclear how they shall best be implemented, and which ones truly matter to users. An interesting case is the existence of pages. The only reason why pages exist in paper books is as a practical arrangement to avoid the inconvenience of books on a single sheet of several squared meters. With digitized books, it becomes possible to ignore pages and simply scroll down from the first to the last line, and bookmark lines, or words, rather than pages. Yet pages remain in eBooks. This is a typical example of a discrepancy between what is possible and what the user wants (or rather, in this case, what the user is not ready to give up).

In addition, the fact that the books of our collection are digitized and not digitally-born (unlike, e.g., eBooks) brings up further challenges. Whereas a lot of data is readily available in digitally-born books, it actually needs to be acquired for digitized books. The state of the art in OCR technologies is rarely a problem for aquiring the textual content. However, the acquisition of the books' logical structure remains very problematic; In Section 4.3 where we describe the set up of a competition and an evaluation framework to address this problem.

4.1.3.2 New approaches at reach

While new research questions came up, a number of existing ones may also be looked at with a different perspective. Such questions have already been addressed, but the availability of books in digital format allows for new approaches.

A number of applications may correspondingly be revisited, notably in library science and information retrieval.

One simple cause for this is that all NLP and IR applications may now be based on the full content of books. Priorily, they were based on summaries, reviews, or back-of-book indexes manually built by librarians.

One might then assume, e.g., that the performance results of document retrieval and document classification shall naturally improve when given the full text (or will it?). Numerous simple questions need to be asked again.

The task of constructing back-of-book indexes is one that shall naturally be automated with standard feature selection techniques. But how will that compare to the work of librarians? Which one of those shall be most useful to readers? Which one of those shall be most useful to an IR system?

Another question is whether or not the back-of-book index can actually support document retrieval, and Wu et al. [WKT08] actually experimented with back-of-book indexes and hinted that they permit an improvement the performance of traditional document retrieval over book collections. An remaining subsequent question is then to evaluate the impact of manually built back-of-book indexes versus the impact of automatically built indexes.

Recommendation systems are another research topic to be revisited. In general, book recommendations have been based on the analysis of transaction data (people who bought A bought B in 80% of transactions, therefore, since you are interested in A, you may be interested in B). Being able to integrate the content of books into such systems would be a definite plus, as most specificities of books are currently ignored.

Another crucial question posed by the collection is that of *scalability*. While Web search engines have already permitted the development and testing of scalable methods for **large collections** of documents, the book collection brings a similar question for collections of **large documents**.

4.1.3.3 Questions to be revisited

Finally, another type of research questions potentially triggered follows the verification of common assumptions. It is possible that a number of approaches based on best practice do not hold with large book collections.

For instance, it is not guaranteed that a number of general assumptions of IR still hold in the context of book data. Indeed, most of the domain's results are based on much shorter documents; the Reuters-21578 collection [Reu87] of financial news stories has been the basis of most early research in document classification, and the domain evolved to take into account collections of scientific articles and the Web.

There has been little documented experimentation with documents as large as books, meaning a number of "rule of thumb" approaches may be questioned, in particular when it comes to aspects that may be impacted by the size of documents: document length normalization [Sin97] and feature selection are typical examples; Should the selection be tightened in ordered to accomodate for larger documents, with many more features, or are the current techniques adequate as is?

In short, a broad research question is to find out whether the best practice of IR remains best practice when dealing with books.

This is especially true in the subfield of structured information retrieval (SIR) studied at INEX. While the same new challenges brought by the size of the document collection and the size of the documents are strong motivation for further research, the need to validate the techniques of SIR is especially strong since the latest results are fairly recent, and were based on smaller collections of much smaller documents.

In my humble opinion, book collections may actually be the key type of data to validate and justify research in SIR. Document retrieval (DR) has indeed been mostly sufficient to search the earlier INEX collections (Wikipedia, scientific articles). There, SIR has allowed fine-grained retrieval of document parts, but the added-value of SIR vs. DR remained fairly limited, since reading or navigating through a whole document was rarely a difficulty. However, in many use cases of book retrieval, returning documents (i.e., books) is definitely a hindrance, and this may be one test case where SIR is a necessity rather than only an interesting addition.

Throughout the years, the Book track evolved, offering tasks in the field of (structured) information retrieval, human-computer interaction, document analysis, and more. The following subsection gives a summary of the tasks, sorted by research domain and by year.

4.1.4 Created Tasks: Chronology and Taxonomy

The ways the book collection shall be explored are numerous, in part because it is the first of its kind to be made available to the community. This section describes the evaluations that we put in place since we distributed the collection in 2007. The chronology of the tasks is summarized in Figure 4.3.

- 2007 : Book Retrieval Task, Page in Context, Classification Task (LoC), User Intent Taxonomy (*Gabriella Kazai and Antoine Doucet*).
- 2008 : Book Retrieval Task, Page in Context, Structure Extraction (new), Active Reading (**new**) (*Monica Landoni joined*).
- 2009 : Book Retrieval Task, Focused Book Search (**new**), Structure Extraction, Active Reading (*Marjin Koolen joined*).
- 2010 : Best Book to Reference (**new**), Prove It (**new**), Structure Extraction, Active Reading (*Jaap Kamps joined*).
- 2011 : Social Search for Best Books (**new**), Prove It, Structure Extraction, Active Reading (The book collection becomes the official collection of INEX).

Figure 4.3: A quick chronology of the task offered in the book track (2007–2011)

The rest of this section will summarizes the various tasks we offered in each of the following domains of research: (structured) information retrieval, document analysis and human-computer interaction. The methodologies developed in (structured) information retrieval and in document analysis are respectively the focus of Section 4.2 and Section 4.3.

4.1.4.1 (Structured) Information Retrieval

Virtual bookshelf, a.k.a. Classification Task (2007). By displaying related books in proximity of each other on book shelves, libraries supports serendipitous browsing and discovery. Motivated by this, we propose a task to build virtual book shelves using content-based IR techniques, such as classification. Participants are required to create and submit a fixed length list of books related to a given user query. The evaluation of the corresponding virtual book shelves is conducted through user tasks based on the queries and observing users' browsing behaviour and collecting judgement on the usefulness of the related books presented to them. Analysis of the way users browse a virtual book shelf and how successful they are in completing their task is used to provide quantitative evaluation.

The Book Retrieval Task (2007–2009). This task was a straight-forward document retrieval task, applied to the book corpus.

The Page in Context (2007–2008) and Focused Book Search Task (2009) The goal of this task was to investigate the application of focused retrieval approaches to a collection of digitized books. The task was thus similar to the INEX ad hoc track's Relevant in Context task, but using a significantly different collection while also allowing for the ranking of book parts within a book. The user scenario underlying this task was that of a user searching for information in a library of books on a given subject, where the information sought may be 'hidden' in some books (i.e., it forms only a minor theme) while it may be the main focus of some other books. In either case, the user expects to be pointed directly to the relevant

book parts. Following the focused retrieval paradigm, the task of a focused book search system is then to identify and rank (non-overlapping) book parts that contain relevant information and return these to the user, grouped by the books they occur in.

The Page in Context task preceded the Focused Book Search Task in 2007 and 2008, with the restriction that the only book parts that may be returned were pages (or lists of pages).

The Best Books to Reference (BB) Task (2010). This task was set up with the goal to compare book-specific IR techniques with standard IR methods for the retrieval of books, where (whole) books are returned to the user. The user scenario underlying this task is that of a user searching for books on a given topic with the intent to build a reading or reference list, similar to those appended to an academic publication or a Wikipedia article. The reading list may be for research purposes, or in preparation of lecture materials, or for entertainment, etc.

The Prove It (PI) Task (2010). The goal of this task was to investigate the application of focused retrieval approaches to a collection of digitized books. The scenario underlying this task is that of a user searching for specific information in a library of books that can provide evidence to confirm or reject a given factual statement. Users are assumed to view the ranked list of book parts, moving from the top of the list down, examining each result. No browsing is considered (only the returned book parts are viewed by users).

Social Search for Best Books (2011). Building on a new collection from Amazon Books and LibraryThing.com, this task investigates the value of user-generated metadata, such as reviews and tags, in addition to publisher-supplied and library catalogue metadata to aid retrieval systems in finding the best, most relevant books for a set of topics of interest. Systems need to return a reading list comprising a ranking of recommended best books for each topic. Topics vary in their format, from simple queries to more extensive descriptions of the information need, to include example books and indications of the user's level of knowledge on the topic.

4.1.4.2 Document Analysis

Structure Extraction (2008–2011). Current digitization and OCR technologies produce the full text of digitized books with only minimal structure information. Pages and paragraphs are usually identified and marked up in the OCR, but more sophisticated structures, such as chapters, sections, etc., are not recognised. Such structures are however of great value in supporting searchers and readers to navigate inside digital books. The task is to build hyperlinked table of contents for a sample collection of digitized books of different genre and style.

4.1.4.3 Human-Computer Interaction

Active Reading (2008–2011). Building on a selection of up to 50 books, relevant to selected user communities (e.g., children establishing their literacy skills, historians, adults reading for pleasure etc.) in specific scenarios (e.g., fact finding,

learning, reading for pleasure, etc.), this task involves conducting a series of user studies into active reading, exploring how and why readers use eBooks with a focus on eBook usability. Participants share a common study design and will be supported in preparing a suitable collection, setting up the studies and analysing the resulting data. The main outcome of this task is the comparison of results across collections and scenarios.

4.2 Evaluating Book Retrieval

In this section, we will describe our work in setting up a book retrieval (BR) evaluation framework within the Book Search track of INEX. Naturally, we will focus on the specific research challenges posed in the field of the *evaluation* of BR, not on the challenges of retrieval themselves, since those were rather addressed by participants.

4.2.1 Problem Description

At first glance, it may not be obvious that the existing IR methodology needs to be accommodated to the case of book retrieval. However, a number of features of book collections have implied a necessary review of the general IR evaluation process. The main practical issues of the evaluation process are simply due to the size of the collection, and the size of its documents.

INEX. A framework for the evaluation of structured document retrieval was built through the INEX initiative from 2002 to 2007. It initiated the idea of a resource-free and collaborative process for document pooling and annotation.

However, the massive change of scale, from collections of scientific articles to full digital libraries requires to revisit the whole IR evaluation framework. The massive increase in terms of recall base implies that the general assumption that "most" relevant documents where found in the top N answers of at least one of the participating systems may not hold anymore. Additionally, the task of verifying whether document fragments are relevant or not is grueling, when the documents are entire books. Subsequently, the task of exhaustively running human assessors through the collection to mark all relevant passages is simply unachievable. The rest of this section will explain why.

Size matters to annotations.

Recall and precision are key evaluation metrics in document retrieval. It is important to remember their definition to better understand the impact of document size.

Precision is the proportion of returned documents which are indeed relevant, while *recall* is the proportion of relevant documents found by the system, out of all the relevant documents in the collection.

Clearly, using these two measures requires to know which documents are relevant and which ones are not. This is usually done through the process of manually annotating document relevance (with respect to a given query). This process is widely described in the literature [Rij79].

Annotations and pooling. A part of the annotation process that is often overlooked is that of the pooling of the annotation set. It is very unrealistic to expect to browse through an entire document collection to mark all its relevant documents (or document parts in the case of structured IR), and do that with respect to each given document query. Hence an important prior of the annotation process is to determine which documents will be annotated, and which can be deemed irrelevant without a manual check. In evaluation efforts, such as TREC[5], the most widely used technique consists in first gathering ranked lists of documents submitted by various systems, and then including the top n documents of each submission into the pool of documents to be annotated, as was originally suggested by Spärck Jones and Van Rijsbergen [SJVR75].

Naturally, this means that a number of relevant documents will be mistakenly counted as irrelevant. When it comes to precision, this is hardly a problem, especially for the evaluation of participating systems (those whose submissions were included in the pooling phase), since it is guaranteed that their top answers are annotated.

When it comes to recall, however, it is very hard to estimate how many relevant documents were wrongly assumed to be irrelevant, because they were not returned in the top n answers of any system.

Recall base. As we have seen, the recall value of a submission is the percentage of all the relevant documents that it contains. The recall base is the total number of relevant documents in the collection. However, this number is almost always wrong, and it is assumed that it diverges even more strongly from reality when the document collection is large.

It is important to underline that the use of a wrong recall base does not matter much when it comes to comparing systems that were used in the pooling phase. Indeed, in such a situation, while the recall values are overestimated, the ranking of systems would be unchanged with an exhaustive recall base.

However, the real issue resides in the later use of the document collection, after the annotation set was compiled. Assuming an homogeneous set of participating systems (systems using similar techniques, hence obtaining similar results) and a very distinct system to be estimated *a posteriori*, it is a matter of fact that all of its original results (those not in the pool) will be counted as irrelevant, whatever the opinion of the annotator would have been.

Specificities in Structured IR. For simplicity, this section mostly dealt with the evaluation of document retrieval. The main difference in structured IR is that, rather than annotating a documents as a whole, annotators are asked to go through the whole document and highlight all the *parts* that are relevant.

The evaluation metrics are naturally more sophisticated, since they must take into account the varying overlap between the document parts in the system submissions, and the document parts that were annotated as relevant. Hence, the main measures deal with notions of "partial relevance" and handle the fact that several document parts of a submission may overlap (making sure that a same relevant document fragment is not counted more than once) [KL06].

[5]The Text REtrieval Conference (TREC), `http://trec.nist.gov/`

Annotating ONE document is a challenge. Unlike at TREC for instance, there are no resources assigned to the annotation process at INEX. Since 2002, the yearly annotation effort have been the result of the participants' collaborative work.

The annotation process is known to be grueling, and this is the reason for the pooling process. However, when it comes to books, we are actually reaching a point when asking to annotate **one** document is unrealistic. This means to ask the annotator to read a book and mark all relevant passages with respect to a given query. In addition, for the consistency of annotations, it is widely recommended that the full annotation of a query be done by one and same annotator (this procedure has always been observed at INEX).

As the collection includes tens of queries yearly, and contains about 50,000 books, the challenge at hand seems unbearable, even with an extremely small annotation pool.

Consequently, the development and test of an annotation system in 2007, through the "traditional INEX way" (that is, collaborative annotations by participants based on the pool of retrieved results), was soon followed by experiments opening the annotation to "the crowd".

4.2.2 Contributions

Annotating book collections. The scalability issue made it necessary to review the annotation process and find innovative ways to annotate book collections. Such are our contributions to the evaluation of Book Search, described in this section. We mostly experimented with game theory and crowdsourcing.

In this section, we will describe the approaches we developed to allow for an ever greater number of reliable, manual annotations. We naturally include results validating the approaches.

4.2.2.1 Setting up the evaluation framework

During the first years of the book search track (2007-2008), we developed the framework to allow for the evaluation of book retrieval, initially providing the tools for a straightforward "standard" evaluation of IR applied to book collections [25]. This included the preparation of a clean collection, publicizing the competition, scheduling, distributing the collection, providing an annotation platform and evaluation software as well as giving out detailed guidelines. All these tasks are grueling but most of them belong to the domain of management rather than to that of original research. The most interesting element is the Book Search System, which evolved throughout the years and remains an essential tool of the evaluation procedure.

The Book Search System, developed at Microsoft Research Cambridge, is an online web service that allows participants to search and browse the books in the Book Track corpus. It is available publicly at `http://www.booksearch.org.uk`.

The Book Search System provides a complete relevance assessment module, which allows users to annotate books and pages inside the books, adding for example relevance labels.

Screenshots of the system are shown in Figures 4.4 and 4.5. Figures 4.4 shows the list of books (assessment pool) to be judged for a given topic (selected by the user). The list was built by pooling the submitted runs (using a round robin process)

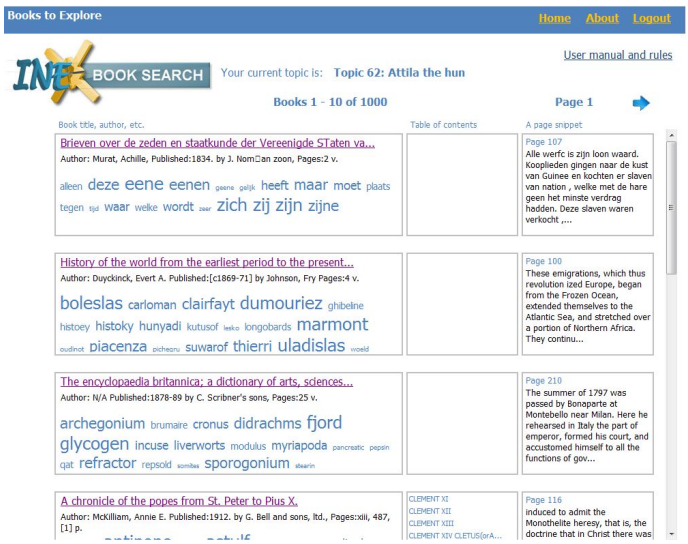

Figure 4.4: Screenshot of the relevance assessment module of the Book Search System: List of books in the assessment pool for a selected topic.

and merging additional search results from the Book Search System itself. On accessing a book, the book is opened in the Book Viewer window (Figure 4.5). There, users can browse through the book and search inside it, or go through the pages listed in the Assessment tab. The pages listed there were extracted from the Page in Context runs. Users can highlight text fragments on a page by drawing a highlight-box over the page image. They can mark whole pages or a range of pages as relevant/irrelevant. Users are also asked to rate the relevance of the whole book. A detailed user manual and system description is available at `http://www.booksearch.org.uk/BECRulesAndUserManual.pdf`.

However, as we discussed extensively, the method of sharing the annotation load amongst participants, as is the tradition in INEX, did not function well for books. In the first years, one issue is that in spite of considerable interest (27 institutions signed up in 2007, 54 in 2008), very few participants actually managed to tame the book collection and submit runs. Because the number of participants and the number of annotators is linked, it was very unlikely to collect sufficient relevance annotations. Adding to that the fact, discussed earlier, that annotating a book collection is intrinsically most costly, it became clear that something needed to be done to increase the amount of annotations gathered, while ensuring their quality.

To do this, there are two ways to proceed, ideally used concurrently. One of them is to increase the motivation of the annotators which we did through gaming, and the other is to increase their number which we did through crowdsourcing.

4.2.2.2 Triggering more assessments through gaming

We motivated annotators by creating an "annotation game" in the Book Track 2008 [20, 22], where annotators compete with each other in both terms of quantity and quality of their annotations.

Indeed, the development of the system and the challenge of scale coincided with works of Luis Van Ahn of Carnegie Mellon, who coined the concept of "human

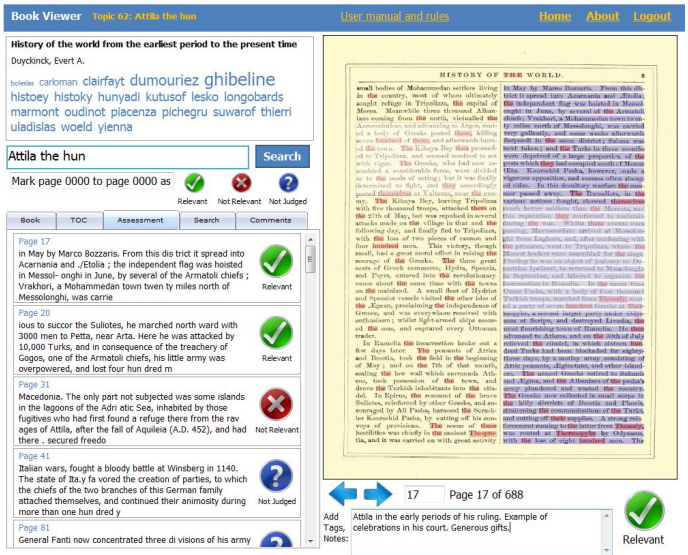

Figure 4.5: Screenshot of the relevance assessment module of the Book Search System: Book Viewer window with Assessment tab listing pages to judge.

computation". He developed the ESP game in 2004 [AD04], in which the goal of each player is to label images with as accurate words as possible. Two players are partnered and shown the same image, and they score points for each word label that they write in common. Since the only thing the two partners have in common is that they both see the same image, they must enter reasonable, consensual labels to have any chance of agreeing on one. The surprising amount of data collected gathered attention from the community, as well as from the industry since Google bought a licence to create its own version of the game in 2006, in order to gather annotations to help improve its image search system.

This work was naturally great inspiration to the idea of games around the gathering of relevance assessments from books.

BookSearch'08 : the Book Explorers' Competition [22]. The relevance assessment system has been made available publicly as part of an online competition called the Book Explorers' Competition, where anyone interested may register and compete for prizes sponsored by Microsoft Research. The competition involves reading and marking relevant content inside books for which users are rewarded points.

For the collection of relevance assessments, a game called the Book Explorers' Competition was therefore designed and deployed in 2008 at Microsoft Research Cambridge under the lead of Gabriella Kazai [KMFC09]. In this competition, assessors (as individuals or as members of teams) competed for prizes sponsored by Microsoft Research. The competition involved reading books and marking relevant content inside the books for which assessors were rewarded points. Assessors with the highest scores at the close of the competition were pronounced the winners. The game was modeled as a two-player game with competing roles: explorer vs. reviewer. An explorer's task was to judge the set of pooled pages as well as to locate and mark additional relevant content inside books. Reviewers then had the task of checking the quality of the explorers' work by providing their own relevance assessments for each page that has been judged by at least one explorer. During

this process, the reviewers could see the relevance assessments of all the explorers who assessed a particular page. In addition to the passage level exploration, both explorers and reviewers were required, independently (information was not shared), to assign a degree of relevance to the book as a whole (on a scale from 0 to 5, with 5 designating the highest degree of relevance).

In total, 3,674 unique books and 33,120 unique pages were judged across 29 topics by 17 assessors. In other words, the unique book per assessor ratio amounted to 216 and the unique page per assessor ratio was 1,948. Such numbers are very satisfying considering the effort needed to assess the relevance of just one page.

BookSearch'09 : Read and Play [20]. In 2009, a new version of the game was set up. Based on what we learnt in 2008, we modified the game to consist of three separate, but interconnected 'Read and Play' games: In game 1, participants had the task of finding books relevant to a given topic and then ranking the top 10 most relevant books. In game 2, their task was to explore the books selected in game 1 and find pages inside them that are relevant to a given topic aspect. Finally, in game 3, their task was to review pages that were judged in game 2. Hence, we had, in essence, introduced a filtering stage (game 1) before the Book Explorer's Competition (game 2 and 3) in order to reduce the number of books to judge in detail.

We ran the 'Read and Play' games for three weeks (ending on March 15, 2010), with weekly prizes of $50 worth of Amazon gift card vouchers, shared between the top three scorers, proportionate to their scores. Additional judgments were collected up to the period of April 15, 2010, with no prizes.

In total, we collected 4,668 book level relevance judgements from 9 assessors in game 1. Assessors were allowed to judge books for any topic, thus some books were judged by multiple assessors. The total number of unique topic-book pair judgements is 4,430.

It is clear that game 1 proved much more popular than games 2 and 3. There are two principle reasons for this. On the one hand, games 2 and 3 can only start once filtered through to them from game 1. On the other hand, in game 1, it is enough to find a single relevant page in a book to mark it relevant, while in games 2 and 3, judges need to read and judge a lot more of a book's content.

Out of the 4,430 books 230 was judged by 2 assessors and 4 by 3 judges. Judges only disagreed on 23 out of the 230 double-judged books, and 2 of the 4 triple-judged books.

This problem has been addressed through the use of crowdsourcing.

4.2.2.3 Triggering more assessments through crowdsourcing

Because the amount of data to be annotated is tremendous, one way to deal with the annotation process may be to get a tremendous number of annotators. This is why we experimented with crowdsourcing during Book Search 2010 [18]. By harnessing the collective work of the crowds, crowdsourcing offers an increasingly popular alternative for gathering large amounts of data feasibly, at a relatively low cost and in a relatively short time.

We experimented with Amazon's Mechanical Turk (AMT) service to aid in the creation of topics for the test collection, and to collect relevance judgements, where

the significant effort required is otherwise inhibiting.

To this end, we first redefined the search tasks, simplifying them in order to make topic creation and relevance assessments suitable as Human Intelligent Tasks (HIT), the basic task units to be offered on AMT for a marginal amount of money, paid out to assessor.

Naturally, the main concern is then trust: can we be sure that annotators truly annotate, and that they do not simply "click around" to simulate relevance judgments and get paid?

To evaluate the reliability of the AMT relevance judgments, the annotation process was first run in the usual way, based on the work of INEX participants as detailed earlier.

Then, in a second phase, 21 HITs were created (one for each topic), consisting of 10 pages to judge, where at least one page was already labeled as confirm or refute by an INEX participant. This was done to ensure that a worker encountered at least one relevant page and that we had at least one label per HIT to check the quality of the worker's work.

Analysis of Crowdsourced Relevance Labels. To verify the consistency of the annotations of AMT workers, each annotation done by one of them was also assigned to two others. The outcome was a consensus of 0.90, which means that, on average, the majority vote for a label forms 90% of all worker votes. If we consider only binary labels, the percentage agreement is higher. Also the agreement among the different degrees of relevance is high with 0.78.

We also look at agreement between the relevance judgments derived from the majority vote of the AMT labels with gold set of INEX labels. Agreement over all 4 label classes is 0.72. AMT workers are more likely to label a page as refute or confirm than INEX participants, which is natural since they are always given one such label within every 10 pages.

The outcome of the analysis of the crowdsourced relevance labels is very positive and paved the way to completely removing the burden of relevance assessments from the participants in future rounds of the book search track.

4.3 Evaluating Book Structure Extraction

As we have seen, one major limitation of the book corpus is the fact that its structure is physical, rather than logical. Following this, the evaluation and relevance judgements based on the book corpus have essentially been based on whole books and selections of pages. This is unfortunate considering that books seem to be the key application field for structured information retrieval, and the fact that for instance, chapters, sections, and paragraphs, are not readily available has been a frustration for the structured IR community gathered at INEX, because it does not allow to test the techniques created for collections of scientific articles and for the Wikipedia.

Unlike digitally-born content, the logical structure of digitized books is not readily available. A digitized book is often only split into pages with possible paragraph, line and word markup. This is also the case for our 50,000 digitized books collection. The use of more meaningful structure, e.g., chapters, table of contents, bibliography,

or back-of-book index, to support focused retrieval has been explored for many years at INEX and has been shown to increase retrieval performance [ZL07].

Mass-digitization projects, such as the Million Book project[6], efforts of the Open Content Alliance[7], and the digitization work of Google[8] are converting whole libraries by digitizing books on an industrial scale [Coy06]. The process involves the efficient photographing of books, page-by-page, and the conversion of each page image into searchable text through the use of optical character recognition (OCR) software.

Current digitization and OCR technologies typically produce the full text of digitized books with only minimal structure information. Pages and paragraphs are usually identified and marked up in the OCR, but more sophisticated structures, such as chapters, sections, etc., are currently not recognized. In order to enable systems to provide users with richer browsing experiences, it is necessary to make available such additional structures, for example in the form of XML markup embedded in the full text of the digitized books.

To encourage research aiming to provide the logical structure of digitized books, we created the book structure extraction competition, which we later brought to the community of document analysis.

Starting from 2008, within the second round of the INEX Book Track, we entirely created the methodology to evaluate the structure extraction process from digitized books: problem description, submission procedure, annotation procedure (and corresponding software), metrics and evaluation. All this is described in the current section. While Gabriella Kazai led the evaluation of Book Retrieval, I was the person in charge for the work in the evaluation of Structure Extraction.

4.3.1 Context and Motivation

The overall goal of the INEX Book Track is to promote inter-disciplinary research investigating techniques for supporting users in reading, searching, and navigating the full texts of digitized books and to provide a forum for the exchange of research ideas and contributions. In 2007, the track focused on information retrieval (IR) tasks [8].

However, since the collection was made of *digitized* books, the only structure that was readily available is that of pages, each page being easily identified from the fact that it corresponds to one and only one image file, as a result of the scanning process. In addition, a few other elements can easily be detected through OCR, as we have seen with the DjVu file format (an example of which was given in Figure 4.1 page 90): this mark-up denotes pages, words (detected as regions of text separated by horizontal space), lines (regions of text separated by vertical space), and "paragraphs" (regions of text separated by a significantly wider vertical space than other lines). Those paragraphs, however, are only defined as internal regions of a page (by definition, they cannot span over different pages).

Hence, there is a clear gap to be filled between research in structured IR, which relies on logical structure (chapters, sections,...), and the digitized book collection,

[6] http://www.ulib.org/

[7] http://www.opencontentalliance.org/

[8] http://books.google.com/

which contains only physical structure. From a cognitive point of view, retrieving book pages may be sensible with a paper book, but it is non-sense with a digital book. The BookML format, of which we gave an example in Figure 4.2 page 91 is a better attempt to grasp the logical structure of books but it remains clearly insufficient.

4.3.1.1 Structured IR requires structure...

In the context of e-readers, even the concept of a page actually becomes questionable; what are pages if not a practical arrangement to avoid printing a book on a single 5 squared meters sheet of paper? For the moment, it seems, however, that users are still attached to the concept of a page [9], mostly as a convenient marker of "where did I stop last?", but when they can actually bookmark any word, line, or fragment of the book, how long will users continue to bookmark pages?

It is important to remember that books as we know them are only a step in the history of reading devices, starting from the papyrus, a very long scroll containing a single sequence of columns of text, used during 3 millenia until the Roman codex brought up the concept of a page. The printing press in the 15^{th} century allowed the shift from manual to mechanical copying, bringing books to the mass [Van99]. Because reading devices, after switching from papyrus to paper, are now living another dramatic change from paper to digital format, it is to be expected that the unnecessary implications of the paper format will disappear in the long run. All physical structure is bound to disappear or come widely unstable. For instance, should pages remain, the page content will vary widely every time the font size is changed, something that most e-readers allow.

What shall remain, however, is the logical structure, whose reason to be is not practical motivations, but an editorial choice of the author to structure his works and to facilitate the readers' access.

Unfortunately, it is exactly this part of the structure that our book collection missed. On the one hand, it seemed to be an ideal framework for structured IR, while on the other, the collection's logical structure was hardly usable. This motivated the design of the book structure extraction competition [10], to bridge the gap between the digitized books and the (structured) IR research community.

4.3.1.2 Context

In 2008, during the second year of the INEX book track, the book structure extraction task was introduced [22] and set up with the aim to evaluate automatic techniques for deriving structure from the OCR texts and page images of digitized books.

The first round of the structure extraction task was "beta"-run in 2008 and permitted to set up appropriate evaluation infrastructure, including guidelines, tools to generate ground truth data, evaluation measures, and a first test set of 100 books that I built. The second round was run both at INEX 2009 [20] and additionally at the International Conference on Document Analysis and Recognition (ICDAR) [13]

[9] see the notes of O'Reilly's Peter Meyers' on his book "Breaking the Page", expected in 2012: `http://newkindofbook.com/`, visited 24 October 2011

[10] `http://www.info.unicaen.fr/~doucet/StructureExtraction/`

where it was accepted as an official competition. This allowed to reach the document analysis community and bring a bigger audience to the effort whilst inviting competitors to present their approaches at the INEX workshop. This further allowed to build up on the established infrastructure with an extended test set and a procedure for collaborative annotation that greatly reduced the effort needed for building the ground truth. The competition was run again in 2010 at INEX [18] and in 2011 at ICDAR [13] (INEX runs every year whilst ICDAR runs every 2^{nd} year).

In the next section, we will describe the full methodology that we put in place from scratch to evaluate the performance of Book Structure Extraction systems, as well as the challenges and contributions that this work involved.

4.3.2 Setting up a Competition

The goal of the competition is to evaluate and compare automatic techniques for deriving structure information from digitized books, which could then be used to aid navigation inside the books.

More specifically, the task that participants face is to construct hyperlinked tables of contents for a collection of digitized books. As the name of the "structure extraction competition" suggests, the long term goal of this effort is to extract the whole logical structure of documents, but the extraction of ToCs has been planned as a significant milestone, unexpectedly difficult to reach. The next steps will be discussed in the perspectives in Section 4.5.

To evaluate the quality of extracted ToCs, we had to construct an appropriate book collection, define a format for Tables of Contents (ToCs), define metrics to compare extracted ToCs to a ground truth, and last but not least, define ways to build such a ground truth in a reasonable time, while still constructing a ground truth that is large enough to allow for significant results, but without compromising quality and consistency...

4.3.2.1 Defining the corpus

In 2009 and in 2011, the corpus consisted of distinct 1,000 book subsets of the BookSearch track's 50,239 book corpus. Therefore, it consisted of books of different genre, including history books, biographies, literary studies, religious texts and teachings, reference works, encyclopedias, essays, proceedings, novels, and poetry.

To facilitate the separate evaluation of structure extraction techniques that are based on the analysis of book pages that contain the printed ToC versus techniques that are based on deriving structure information from the full book content, we always selected 200 books that did not contain a printed ToC into the total set of 1,000. To do this, we used the BookML format where pages that contain the printed ToC (so called ToC pages) are explicitly marked up. We then selected a set of 800 books with detected ToC pages, and a set of 200 books without any detected ToC pages into the full test set of 1,000 books. Please note that this ratio of 80:20% of books with and without printed ToCs is proportional to that observed over the whole corpus of 50,239 books.

4.3.2.2 Sample Research Questions

To motivate the community of researchers, we provided a sample of open research questions that the competition shall help to address. Example research questions whose exploration is facilitated by this competition include, but are not limited to:

- Can a ToC be extracted from the pages of a book that contain the actual printed ToC (where available) or could it be generated more reliably from the full content of the book?
- Can a ToC be extracted only from textual information or is page layout information necessary?
- What techniques provide reliable logical page number recognition and extraction and how logical page numbers can be mapped to physical page numbers?

4.3.2.3 Task Description

Given the OCR text and the PDF of a sample set of 1,000 digitized books of different genre and style, the task was to build hyperlinked tables of contents for each book in the test set. The OCR text of each book is stored in DjVu XML format (see once more Figure 4.1 page 90). Participants could employ any techniques and could make use of either or both the OCR text and the PDF images to derive the necessary structure information and generate the ToCs. Giving the possibility to use OCR text (DjVu format) was meant to facilitate access from participants with no experience of OCR, and let them start from a preprocessed common-ground format.

Participating systems were expected to output an XML file (referred to as a "run") containing the generated hyperlinked ToC for each book in the test set. The document type definition (DTD) of a run is given in Figure 4.6.

4.3.2.4 Annotation of ToCs : Methodology and Software

Naturally, to compare the submitted runs to a ground truth necessitates the construction of such a ground truth. Given the burden that this task may represent, we chose to split it between participating institutions, and rather than forcing participants to do annotations (which may trigger hasty and careless work), we encouraged them with an incentive: we limited the distribution of the resulting ground truth set to those who contributed a minimum number of annotations. This is pretty much inline with INEX habits. However, placing the burden on participants is evidently a hindrance and the effort must be as limited as possible. This section describes the ground truth annotation process we designed and its outcomes.

Annotation Process.

The process of manually building the ToC of a book is very time-consuming. Hence, to make the creation of the ground truth for 1,000 digitized books feasible, we resorted to 1) facilitating the annotation task with a dedicated tool, 2) making use of a baseline annotation as starting point and employing human annotators to make corrections, and 3) sharing the workload.

```
<!ELEMENT bs-submission
    (source-files, description, book+)>
<!ATTLIST bs-submission
    participant-id  CDATA  #REQUIRED
    run-id  CDATA  #REQUIRED
    task  (book-toc) #REQUIRED
    toc-creation (automatic |
      semi-automatic) #REQUIRED
    toc-source (book-toc | no-book-toc |
      full-content | other) #REQUIRED>
<!ELEMENT source-files EMPTY>
<!ATTLIST source-files
    xml  (yes|no) #REQUIRED
    pdf  (yes|no) #REQUIRED>
<!ELEMENT description (#PCDATA)>
<!ELEMENT book (bookid, toc-entry+)>
<!ELEMENT bookid (#PCDATA)>
<!ELEMENT toc-entry(toc-entry*)>
<!ATTLIST toc-entry
    title (#PCDATA) #REQUIRED
    page  (#PCDATA) #REQUIRED>
```

Figure 4.6: DTD of the XML output ("run") that participating systems were expected to submit to the competition, containing the generated hyperlinked ToC for each book in the test set.

An annotation tool was specifically designed for this purpose and developed at the University of Caen during the traineeship of student Paul Cercueil, under my supervision. The tool takes as input a generated ToC and allows annotators to manually correct any mistakes. A screen capture of the tool is shown in Figure 4.7. In the application window, the right-hand side displays the baseline ToC with clickable (and editable) links. The left-hand side shows the current page and allows to navigate through the book. The JPEG image of each visited page is downloaded from the INEX server at `www.booksearch.org.uk` and is locally cached to limit bandwidth usage.

Using the submitted ToCs as starting points of the annotation process greatly reduces the required effort, since only the missing entries need to be entered. Others simply need to be verified and/or edited, although even these often require annotators to skim through the whole book.

An important side-effect of making use of a baseline ToC is that this may trigger a bias in the ground truth, since annotators may be influenced by the ToC presented to them. To reduce this bias (or rather, to spread it among participating organizations), we chose to take the baseline annotations from participant submissions in equal shares.

Finally, the annotation effort was shared amongst all participants. Teams that submitted runs were required to contribute a minimum of 50 books, while others were required to contribute a minimum of 100 books (20% of those books did not contain a printed ToC). The created ground truth was made available to all contributing

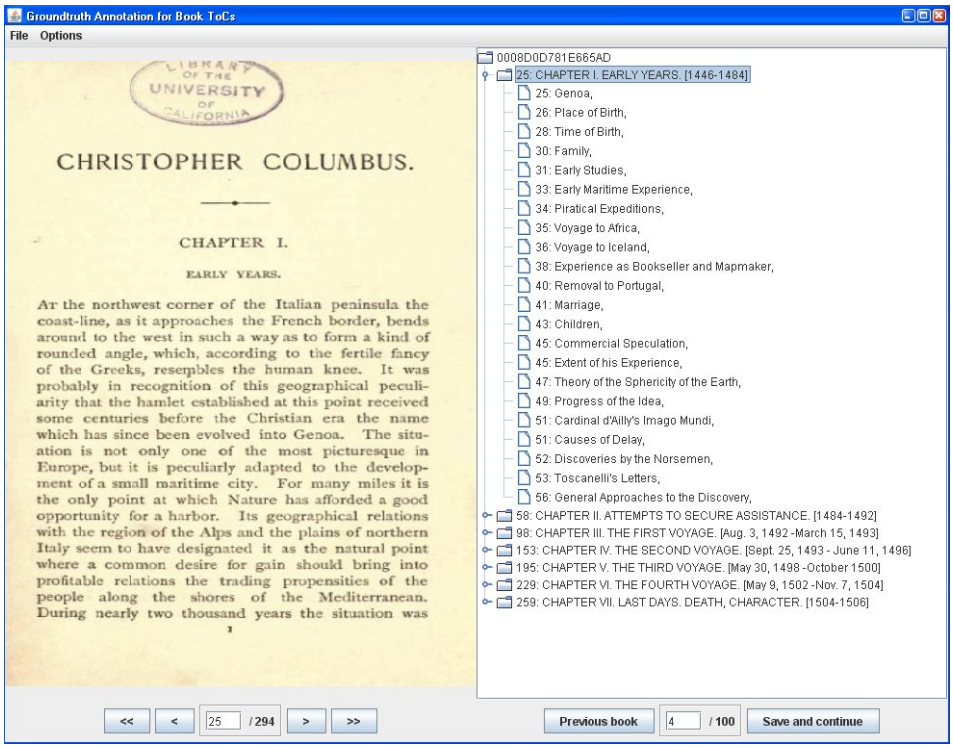

Figure 4.7: A screen shot of the ground truth annotation tool.

participants for use in future evaluations.

Collected Ground Truth Data.

In 2009; Seven teams participated in the ground truth annotation process, 4 of which did not submit runs.

This joint effort resulted in a set of 649 annotated books. To ensure the quality and internal consistency of the collected annotations, each of the annotated ToC was reviewed by the organizers, and a significant number had to be removed. Any ToC with annotation errors were then removed. Errors were most of the time due to failure to follow the annotation guidelines or incomplete annotations.

Following this cleansing step, 527 annotated books remain to form the ground truth file that was distributed to each contributing organization. 97 of the annotated books are ones for which no ToC pages were detected.

In 2011; The output was very similar with 6 teams participating to the annotation phase, 2 of which did not submit run, and a total number of 513 annotated books brought out to form the 2011 ground truth.

4.3.2.5 Validation of the annotation procedure

To validate the methodology, and as the evaluation is based on manually built ground truth, it was crucial to validate the approach by verifying the consistency of the gathered ToC annotations.

To do this, we assigned the same set of books to two different institutions. This resulted in 61 books being annotated twice. We measured annotator agreement by

	Precision	Recall	F-measure
Titles	83.51%	83.91%	82.86%
Levels	74.32%	75.00%	74.04%
Links	82.45%	82.87%	81.83%
Complete except depth	82.45%	82.87%	81.83%
Complete entries	73.57%	74.25%	**73.31%**

Table 4.1: The score sheet measuring annotator agreement for the 61 books that were assessed independently by two distinct institutions.

using one of these sets as a run and the other as the ground truth and calculating our official evaluation metrics (see Section 4.3.2.6). The result of this comparison is given in Table 4.1.

We can observe an agreement rate, of over 70% for complete entries based on the F-measure. It is important to observe that most of the disagreement stems from title matching, which makes us question whether the 20% tolerance utilized when comparing title strings with the Levenshtein distance may need to be increased, so as to lower the impact of annotator disagreement on the evaluation results. However, this requires further investigation as an excessive increase would lead to uniform results (more duly distinct titles would be deemed equivalent).

4.3.2.6 Metrics

The automatically generated ToCs submitted by participants were evaluated by comparing them to a manually built ground truth. The evaluation required the definition of a number of basic concepts:

Definitions. We define the atomic units that make up a ToC as ToC Entries. A ToC Entry has the following three properties: *Title*, *Link*, and *Depth Level*. For example, given a ToC entry corresponding to a book chapter, its *Title* is the chapter title, its *Link* is the physical page number at which the chapter starts in the book, and its *Depth Level* is the depth at which the chapter is found in the ToC tree, where the book represents the root.

Given the above definitions, the task of comparing two ToCs (i.e., comparing a generated ToC to one in the ground truth) can be reduced to matching the titles, links and depth levels of each ToC entry. This is, however, not a trivial task as we explain next.

Matching Titles. A ToC title may take several forms and it may only contain, e.g., the actual title of a chapter, such as "His Birth and First Years", or it may also include the chapter number as in "3. His Birth and First Years" or even the word "chapter" as in "Chapter 3. His Birth and First Years". In addition, the title that is used in the printed ToC may differ from the title which then appears in the book content. It is difficult to differentiate between the different answers as all of them are in fact correct titles for a ToC entry.

Thus, to take into account not only OCR errors but also the fact that many similar answers may be correct, we adopt vague title matching in the evaluation. We say that two titles match if they are "sufficiently similar", where similarity is measured based on a modified version of the Levenshtein algorithm (where the cost of alphanumeric substitution, deletion and insertion is 10, and the cost of non-alphanumeric substitution, deletion and insertion remains 1) [Lev66]:

Two strings A and B are "sufficiently similar" if

$$D = \frac{LevenshteinDist * 10}{Min(length(A), length(B))}$$

is less than 20% and if the distance between their first and last five characters (or less if the string is shorter) is less than 60%.

Matching Links. A link is said to be correctly recognized if there is an entry with matching title linking to the same physical page in the ground truth.

Matching Depth levels. A depth level is said to be correct if there is an entry with matching title at the same depth level in the ground truth.

Matching complete ToC entries. A ToC entry is entirely correct if there is an entry with matching title and same depth level, linking to the same physical page in the ground truth.

Measures. For a given book ToC, we can then calculate precision and recall measures [Rij79] for each property separately, and for complete entries. Precision is defined as the ratio of the total number of correctly recognized ToC entries and the total number of ToC entries in a generated ToC; and recall as the ratio of the total number of correctly recognized ToC entries and the total number of ToC entries in the ground truth. The F-measure is then calculated as the harmonic mean of precision and recall. Each of these values was computed separately for each book and then averaged over the total number of books (macro-average).

The measures were computed over the two subsets of the 1,000 books (see Section 4.1.2), as well as the entire test set to calculate overall performance. The two subsets, originally comprising of 800 and 200 books, respectively, that do and do not have a printed ToC, allowed us to compare the effectiveness of techniques that do or do not rely on the presence of printed ToC pages in a book.

Results. For each submission, a summary was provided in two tables, presenting general information about the run as well as a corresponding score sheet (see an example in Table 4.2).

In this manuscript, we decided to leave the results out, since they only stem indirectly from our work, and are instead the fruits of research from the competition participants.

	Precision	Recall	F-measure
Titles	57.90%	61.07%	58.44%
Levels	44.81%	46.92%	45.09%
Links	53.21%	55.53%	53.62%
Complete except depth	53.21%	55.53%	53.62%
Complete entries	41.33%	42.83%	**41.51%**

Table 4.2: An example score sheet summarizing the performance evaluation of the "MDCS" run.

Alternative Measure and Discussion.

Participants were encouraged to propose alternative metrics, and Meunier and Déjean introduced the XRCE link-based measure to complement the official measures with the aim to take into account the quality of the links directly, rather than conditionally to the title's validity [DM10a].

Indeed, the official measure works by matching ToC entries primarily based on their title. Hence the runs that incorrectly extract titles will be penalized with respect to all the measures presented in the score sheet of Table 4.2. For instance, a system that incorrectly extracts titles, while correctly identifying links will obtain very low scores (possibly 0%). The XRCE link-based measure permits to evaluate the performance of systems works by matching ToC entries primarily based on links rather than titles.

The "complete entries" measure, used as a reference in most of this paper is a global, cumulative measure. Because an entry must be entirely correct, i.e., title, link, etc., to be counted as a correct entry, an error in any of the criteria implies a complete error.

While the various measures presented in Section 4.3.2.6 have in common a sensitivity to errors in the titles of ToC entries, the alternative measure in turn is strongly dependent on the correctness of page links.

We do not claim that success with respect to one metric is more important than with another, but believe that the measures presented should be seen as complementary. Depending on the application or situation, one metric may be preferred over another. For example, if navigation is key, then being able to land the user on a page where a chapter starts may be more important than getting the title of the chapter right.

One of our goals in the future is to provide a toolbox of metrics, to be used by researchers enabling them to analyze and better understand the outcome of each of their approaches. The current version of this toolkit is available on the competition's web site[11].

4.3.3 Summary of Contributions

Starting from scratch, we created a complete framework for the evaluation of structure extraction from digitized books.

[11] `http://www.info.unicaen.fr/~doucet/StructureExtraction/`

Unlike book retrieval, where adjustments had to be made to existing information retrieval evaluation techniques, everything had to be done to be able evaluate book structure extraction. Hence, every step of the competition setup is a contribution: the problem definition, the compilation of the collection, the task description and submission procedure, the definition of evaluation metrics, the annotation format and procedure, etc.

Clearly, in a similar fashion as with book retrieval, the most important challenge is (and remains) to be able to annotate such massive collections. Being able to do it was a first challenge that we have managed to address. The next challenge is to increase significantly the amount of collected ground truth. One obvious way, in the light of the latest development in BookSearch, is to rely on crowdsourcing. This is definitely future work.

The set up of this competition, initially sketched and tested on a low-scale in INEX 2008 was validated by the community for the first time when the competition was accepted by the ICDAR 2009 programme committee as a conjoint event between INEX and ICDAR. The contributions were recognized through the publication of papers, the main one being an article in the International Journal of Document Analysis and Recognition (IJDAR) describing the framework in 2010 [2].

However the most crucial acknowledgement is that of participants of the Structure Extraction competition. 19 institutions have expressed interest, 10 have participated to the ground truth creation, and 5 submitted runs so far. This adhesion to the competition is not only supportive but it is also the only way that we could provide a decent-sized collection to the community.

Standalone test data. Indeed, to facilitate the participation of other institutions in the future, it was decided in 2010 to always make available the second to last ground truth set. We then distributed the initial set of 100 ToCs built during the first Book Structure Extraction task at INEX 2008 [22]. Following, as soon as the 2011 competition started, the data collected in 2009 (527 book ToCs) was made available online. Similarly, during the next round of the competition, the 2011 ground truth set will be released (until then, its access remains restricted to sufficient contributors of the 2011 ground truth set).

Effectively, these ground truth sets, distributed together with the document collection and the evaluation software are forming standalone evaluation packages, freely available to the research community[12].

ICDAR competition and INEX Workshop. A strength of the conjoint organization between INEX and ICDAR, is the effective bridging of two communities: the competition is labelled by and presented at ICDAR, but at the same time, participants are invited to write paper presenting their approaches at the INEX workshop, a selection of which have already been published by Springer Lecture Notes in Computer Science within the INEX workshop proceedings (see, e.g., [DURT09, DM09, DM10b, GL10, CLH11]).

The respective ICDAR and INEX schedules facilitate this, since the ICDAR result deadline is around the middle of the year, while the INEX workshop is generally held in December, with paper submission deadlines at the end of October.

[12] `http://users.info.unicaen.fr/~doucet/StructureExtraction/training/`

Future of the Structure Extraction competition. The competition will continue running in the coming years. This was requested by several participants intending to return, as well by several other institutions who were still developing their structure extraction systems at submission time. These groups typically participated in building the ground truth set, and shall be able to submit runs next time.

Another important reason to maintain the competition is evidenced by the current results, indicating that much could still be improved upon, especially in the case of books that do not contain ToC pages. This underlines how much remains to be done in the field of book structure extraction.

The directions of the future rounds of the Structure Extraction competition are discussed within the global perspectives of my research in evaluation, in Section 4.5. Before that, I will summarize my personal publications related to this topic, and put them in context in the following section.

4.4 Related Publications

The general background of the INEX workshop is best summarized in INEX reports [5, 6, 7] published within the SIGIR forum, the biannual publication of the ACM Specific Interest Group on Information Retrieval (SIGIR). The evolution of the share of the reports dealing with the book track is a good indicator of its growing importance within the INEX framework.

The initial set up of the book retrieval task was presented within INEX 2007 [25], while a more extensive standalone description was published in the SIGIR Forum in 2008 [8]. A number of potential new user tasks were exposed in a short position paper presented at the European Conference on Digital Libraries (ECDL) 2008 [15]. This is where the idea of the structure extraction was proposed for the first time. The later rounds of the book track introduced new tasks as well as variations of existing ones. All these tasks, as well as the participants' approaches, are described in the corresponding book track overviews [18, 20, 22].

The structure extraction competition is also briefly overviewed within each of these papers. However, extensive description and discussion is rather found in publications of the document analysis community. Indeed, following the acceptance of the 1^{st} and 2^{nd} Structure Extraction competitions at the International Conference on Document Analysis and Recognition (ICDAR), respectively in 2009 and 2011, their set ups, overviews and results were published in corresponding ICDAR proceedings [10, 13]. Our contribution to the evaluation of book structure extraction was most extensively described in a longer article, published in the International Journal of Document Analysis and Recognition (IJDAR) in 2010 [2].

In addition, selected papers describing participant approaches are published yearly within the INEX workshop proceedings by *Springer*, as different volumes of the Lecture Notes in Computer Science (LNCS) series [FKLT08, GKT09, GKT10, GKST11]. These volumes contain descriptions of participant approaches to both the book search and the structure extraction task.

4.5 Conclusion and Perspectives

Starting from the distribution a digitized book collection in 2007 and the corresponding very first Book Search track run at INEX, progress has been steady. We have designed techniques to gather a sufficient number of relevance assessments and evaluate Book IR. We also fostered renewed interest and designed evaluation methods for the problem of the extraction of logical structure from digitized books, opening the way for applications of strutured information retrieval in a motivating application setting.

The number of registered participants of the book track has grown from 27 in 2007, to 54 in 2008, and 84 in 2009 and 2010. In 2011, for the 10^{th} anniversary of INEX, the Book track became the main track of INEX, replacing the "*adhoc*" track which evaluated structured IR from with collections of scientific articles (IEEE) and Wikipedia articles, from 2002 to 2010. For both the Book Search the Structure Extraction tasks, participants have been invited to present their approaches at the INEX workshop, with proceedings published by Springer Lecture Notes in Computer Science.

Starting form 2012, the INEX workshop, with the book track as its main track, will be colocated with CLEF, which recently grew from a forum on cross-language evaluation to a full-scale conference focused on Multilingual and Multimodal Information Access.

In addition, it is worth mentioning that the Book track, originating from INEX is gaining external visibility by planting seeds in different areas of information access; the Structure Extraction competition run at ICDAR since 2009 is one example, but a "BooksOnline" workshop [13] has also been run yearly at CIKM since 2008 (except in 2009 when it was located at ECDL). Work on crowdsourcing for relevance annotations has also led Gabriella Kazai to participate in the creation of the TREC Crowdsourcing track in 2011[14] [15].

Future of the tracks. Since participant involvement keeps growing, and the research problems have not been solved, it is natural that the competitions will keep running.

While the Book Search task now offers to explore the exploitation of recommendations and book summaries, partly due to the lack of a logical structure in the book collection, I still keep a strong personal interest in the exploitation of the full content of books, which will be made much easier when the logical structure of books can be properly extracted. To facilitate that, a number of improvements are considered for the future of the Book Structure extraction task.

Crowdsourcing the ground truthing of Book Structure. In spite of the tremendous efforts of participants to build the ground truth, we shall experiment with crowdsourcing methods in the future. This may offer a natural solution to the evaluation challenge posed by the massive data sets handled in digitized libraries.

[13] BooksOnline'11, `http://research.microsoft.com/en-us/events/booksonline11/`

[14] TREC Crowdsourcing track, `https://sites.google.com/site/treccrowd/`

[15] Note: To make sure readers are not mislead, I wish to underline that I am not personally involved in organising neither the BooksOnline workshops, nor the TREC crowdsourcing track.

The step was successfully made in the Book Search task and it is now natural for the Structure Extraction competition to follow a similar path.

Our experience in using crowdsourcing for relevance assessments over the same data set suggests the feasibility of using crowdsourcing reliably in high cognitive tasks such as that of labelling ToCs.

Other Evaluation Techniques. We also aim to investigate the usability of the extracted ToCs. In particular, we will explore the use of qualitative evaluation measures in addition to the current precision/recall measures. This would enable us to better understand what properties make a ToC useful and which are important to users engaged in reading or searching. Such insights are expected to contribute to future research into providing better navigational aids to users of digital book repositories. This effort shall be led through crowdsourcing.

This crowdsourcing proposal and the previous one both offer interesting questions in quality-control, a key issue to make the output of crowdsourcing useful. These shall be relevant internship topics for Master students.

Further Structure

As the name "Book Structure Extraction" competition suggests, tables of contents are not the sole objective, but rather a first milestone, that proves far more difficult to reach than expected.

In the future, however, we plan to expand the task to include the identification of more exhaustive structure information, e.g., header/footer, bibliography, etc. This shall happen in connection through a collaboration with the University of Innsbrück, Austria.

Contacts with the IMPACT project. In Section 4.1, we introduced the 4-year European project IMPACT, started in 2008. Its main overlap with our work resides in the problem of structure extraction, which is the focus of the "Enhancement & Enrichment sub-project" whose goal is "to make the OCR results more accurate and more accessible (...) work on collaborative correction, descriptions of physical and logical structure (...)".

In 2009, I contacted with the research group of the University of Innsbrück, which is in charge of this sub-project. They were very eager to join our effort and offered to provide their manually annotated structure, which is constructed by a sub-contractor (however, none of it was available as of mid 2011). They further provided their annotation tool in 2010. Unfortunately, it appeared too sophisticated for our needs, and crucially, it needed to be downloaded, installed and used locally, which posed a problem w.r.t. our aim of relying on crowdsourcing.

To test the techniques they developed, they participated to the structure extraction competition for the first time in 2011 [10], but unfortunately failed to submit results in time.

In the future, closer collaboration is expected. They notably offered to distribute the results of their ground truthing effort through the competition and their annotation tool. While the annotation tool was not quite suited for our purpose, the ground truth will be a welcome addition, as it contains far more structure that just ToCs.

Book Retrieval Projects

Book retrieval remains essentially unexplored and provides room for considerable doctoral work. The wide range of research questions I listed and categorized in Section 4.1.3 have not been handled and constitute projects I would gladly supervise. All of these are questions I have in mind for a couple of years, unfortunately always lacking the time and resources for serious exploration so far. In particular, the questions related to the application of structured retrieval to book collections are particularly interesting, as I strongly believe that book collections are the key application domain for structured IR.

However, to finally be able to proceed with this work, one of two things is required: 1) better performance of the structure extraction systems or 2) the distribution of a digitally-born, well structured book collection. With the respect to the first option, it is evident that being the organizer of the main event to produce the structure of digitized books shall give me an edge to lead such research. According to private conversation with one the organizers, the second option is expected to emerge within months via the TREC contextual track 2012 [16], where a corpus of slightly outdated Lonely Planet guidebooks shall be distributed for IR research purposes.

Human-Computer Interaction. An important fact about e-readers is that they deprive readers from a lot of context. Being returned only a fragment of text is not the same as being given a pointer into a printed books. The ability to search for keywords within an eBook is depriving readers from context that is intrinsecally available with paper books [CP03]. This poses many questions in HCI that were barely overviewed within INEX.

[16] TREC contextual, `http://sites.google.com/site/treccontext/`

Chapter 5

Conclusion and Perspectives

In this dissertation, I have covered the essential research attempts of my career thus far. Its raw material has been document, under many different forms, and through techniques as generic as possible.

I have presented three main chapters, dealing with the extraction of knowledge, the exploitation of knowledge, and subsequent evaluation methodologies.

Writing a conclusion is always a difficult task. Even more so in this case, when I do not see this document as an end, but rather as a rare opportunity to sit back and wonder where I started, where I am, and where I am heading.

Where I started, and especially where I am, I hope, is now clear to the reader, following the previous pages of this manuscript. Where I am heading has been hinted throughout this thesis, with the development of perspectives relative to each section of the document.

However, those perspectives were mostly describing further steps in the near vicinity of previously visited areas. What I wish to introduce in the rest of this chapter is new destinations that I am willing to explore.

Further Perspectives. While there are very many tasks I can envisage, I will sketch, in the following section, three lines of work in which I am especially eager to investigate. The first one is rather an application field (microblogs, short messages), where I believe that language-independent techniques are especially suited (Section 5.1). The second line of work, described in Section 5.2, concerns the development of a full system taking advantage of structured information retrieval to offer personalised services to users, in the context of a comparative evaluation initiative to be launched shortly. The last line of work (to be presented hereby!) is that of the automatic detection of new word relations in news feed streams, detailed in Section 5.3. This plan is the most advanced since collaboration has already started with the group of Hannu Toivonen at the University of Helsinki and I have submitted a detailed proposal in January 2012 to the ANR (*Agence Nationale de la Recherche*) in application for the funding of the French side of this project.

5.1 New Languages

In several parts of my work, I have applied techniques to text, by regarding it as sequential data, even sometimes treating languages as little more than hexadecimal

codes.

For a number of reasons, I am under the impression that such techniques are gaining importance. First, the increase in written material has caused a decrease in the quality of written language. It is widely observed indeed, that the share of emails that for instance contain typographical errors is ever increasing.

Second, new languages have emerged with text messages (SMS) and microblogs (e.g., *Twitter*, *SinaWeibo* and *QQ*). In every "main language", these services gave birth to new dialects, mixing phonetics (e.g., in the use of digits as in "cu4lunch"[1]) into text compression. These dialects, following for a good part the principle of least effort, have started to spread out. In those written dialects, "'lol" has for instance become a more and more widely accepted written French term, and it is probably only a matter of time until it becomes an acceptable utterance.

New as they are, those dialects tolerate wide variations. There are very many ways to write the same thing, and the only thing that matters is that the recipient gets a chance to decode the message. You may write "see you" as "CU", "C U", "C you" or "see U". In other words, there shall be no typographical errors, because there is no such thing as orthography.

In addition, one might wonder about the future of linguistic tools in such a context, where syntax and orthography matter less and less, and even sometimes do not even exist, in the case of orthography. Interestingly, techniques that ignore grammar, that do not focus on words but rather on characters, that do not remove short words ... suddenly appear very relevant.

To experiment with microblogs and other sources of short messages shall therefore be very adequate for the techniques presented in this dissertation. The applications are numerous, fueled for instance by the interest of corporate and governmental intelligence to mine opinions. A company wishing to set an automated support system accessible by SMS might struggle to interpret user requests.

A lot of work has focused on text messaging, and the corresponding creation and adaptation of existing tools. Meanwhile, our generic approaches were ready to function in the very moment when someone first wrote "CU4lunch". Interestingly, many of the research efforts have focused on translating, e.g. text messages, into the corresponding "main language", so as to later apply existing linguistic tools. This is taking the risk to add up the defects of translation systems to those of the end tools, setting an upper-bound on performance, and necessitating ever more layers within the linguistic process. In other words, to deal with text messages from multiple languages, one needs tools for all those languages, and utilities to translate text messages into their main languages. This process makes multilingual applications even less realistic than in the general case.

However, for global corporations for instance, the possibility to handle multilingual data is especially crucial; being present *worldwide*, they accordingly wish to survey the opinion of their customers *worldwide*. The potential of our methods in this respect needs to be explored, since they are not only multilingual in the sense of the main acknowledged languages, but also in the sense of each of their dialectal variations.

[1] See you for lunch.

5.2 Personalized Contextual Search

In this section, I will introduce a project that is meant to be able to federate interest in a working group on research information, such as the one I am now leading within the Hultech team of the GREYC laboratory. It is thinking of the skills and interests present in the group that I cooked up this project, but, as we will see, it could be easily fitted into different IR research environments.

The main goal of the project is to address the problem of retrieving relevant information for a mobile user in changing contexts, given a vague query, such as "I'm bored, what should I do tonight?". The system shall then exploit all the available contextual and personal information about the user, so as to produce relevant suggestions, in a restricted number of characters (say 140) since we assume that this use case would occur through a mobile device.

We assume that the user has a personal data collection including, e.g. calendar, sent and received email messages, memos, plans, and collections of business cards. The user may also have set a list of taste-based preferences, such as likes and dislikes. Additionally, the retrieval system may have access to other collections in the current environment and on the web. Contrary to traditional information retrieval systems, we do not expect the user to give a well-formulated query. The major trigger for retrieval is the context in which the user is at that moment. The context may consist of all information that is possible to acquire: e.g. time, location, temperature, nearby objects/persons/vehicles/services, or background noise.

A typical example of a context-aware application is a mobile tourist guide. For this scenario, the underlying assumption is that a tourist wants to see and visit places. The tourist can see a bridge if she is located near the bridge. She can visit a museum only if it is open. She may not want to have lunch in an open-air restaurant if it is cold. As we can see, there are many assumptions considering time, location, and temperature in this case. We want to find such common sense assumptions for some other common tasks in working and everyday life and try to generalize them. An example of a generalized assumption regarding time for many scenarios may be that something that is in the past (even if it happened just 5 seconds ago) is not relevant.

A modular project. A first and rapid approach resides in the transformation of personal and contextual facts into query terms, to be compared to document collections before returning snippets of the most relevant documents. However, such a project shall involve a lot more research, and actually each subpart of the project contains numerous open problems: personalization and contextualization, document summarization, (structured) retrieval, opinion mining, social networks ...

Evidently, personalization and contextualization allow for much more sophisticated models. In fact, user modeling is a topic of research of its own (see, e.g., [BSB10, DTB11] for recent approaches).

In addition, suggestions shall not only be based on the likes and dislikes of the users, but they may also be inferred from the likes and dislikes of her social network; If the user is on a distant trip, she may not know that her best friend went to a nearby restaurant that she enjoyed. However, this would be highly relevant information. More globally, given equal personal preferences, our user would prefer to visit places

that are highly regarded rather than despised, and this may be discovered through the opinion mining of microblogs, for instance (relating to the project discussed in the preceding section).

Document summarization is another research problem related to this project. Given that answers are to be given through mobile devices, it is essential to be able to limit their length. One partial approach shall be to expand the techniques presented in Section 3.2 to sentence reduction, i.e., replacing long expressions by shorter semantic equivalents. However, this idea alone would require deep investigation.

As we have seen, this personalized contextual search project can take many shapes, while the end application remains very interesting. In reality, the actual boundaries and focus of the project will result from the list of its participants, their skills, and how much time they have to give.

A relevant evaluation framework. Another way to bound such a project shall be the participation to a comparative evaluation campaign. This further removes the burden of evaluation from the research team.

In February 2012, the details of the first TREC Contextual Suggestion track[2] are expected to be released. As of 6 February 2012, the Website remains almost empty, only stating that the TREC Contextual Suggestion track "deals with complex information needs which are highly dependent on context and user interests".

However, according to private conversation with one of the organizers, the search collection is to consist of a corpus of slightly outdated Lonely Planet guidebooks, distributed for information retrieval research purposes. Such a collection of highly structured books shall offer an ideal setting to perform fine-grained structured information retrieval, and apply the techniques developed through EXTIRP and presented in Sections 3.3 and 3.4.

5.3 News Surveillance

The last discussed perspectives can be seen as the extension of epidemiological surveillance to any domain. However, the technique is fairly different in a number of aspects. Notably, so as to reach a technique that can address any domain without prior assumptions, we are not able to rely on a domain-specific lexicon. This implies that the technique must function without any resources, relying solely on the processed corpus.

From January to March 2012, I am visiting the group of Hannu Toivonen in Helsinki. Based on the idea of cognitive mapping, we are working on a technique to detect novel concept relationships in documents entering a news stream. Such a detection stems from the combination and comparison of *local* association measures (built from the incoming document) and *global* association measures (built from the collection of previous documents).

Relying on the idea of cognitive mapping, our goal is to first build, for a given corpus, a given document and a given term, a *mind map* of all the possible cognitive associations for the term (general associations from the corpus and specific ones from the document). This is meant to allow for the detection of **novelty**. This

[2]TREC contextual suggestion, `http://sites.google.com/site/treccontext/`

differs widely from text mining applications in which knowledge is extracted from a whole (static) corpus. The main application resides in the automatic underlining of novel elements in a news stream, with no need for linguistic resources a priori (hence allowing for multilingual and multi domain results).

The aim of such a "news novelty detector" is to be able, for instance, to automatically detect the surprising co-occurrence of "Dengue" and "Thailand", or "Apple Inc." and "profit warning".

The system would not need to be designed for a specific application domain, but the user would rather fine-tune which application domains she is interested in, in her personal interface.

Originality. The goal of information extraction is to extract certain structured information from textual documents (see, e.g., Cowie and Lehnert [CL96]). Information extraction methods are routinely also used to discover associations between terms. Examples include news story analysis (who did what, where and when) and automatic extraction of biomedical facts from scientific articles (which proteins interact, which gene contributes to which phenotype, etc.). While information extraction methods are tuned to look for specific types of facts (including relations), our goal is to be able to discover *associations* between arbitrary terms.

In topic detection and tracking (TDT) the goal is to recognize events in news stories and relate stories to each other [All02]. Information extraction is one of the key technologies. An example application for our methods is the production of mind maps of news stories. It is largely complementary to topic detection and tracking: the emphasis on relations between terms, both within stories (the novel associations looked for with methods introduced here) as well over several stories (semantic associations in the background). TDT has facilitated the organization of news documents. However, small developments in a news story are not automatically detected and evidenced; only major developments are detected when documents are sufficiently distinct.

The method we propose addresses addresses the problem to find associations that are relatively specific to a document. In a stream, such associations are deemed novel. We already led preliminary experiments with promising results obtained with sentence-based statistics (with a $tpf - idf$ measure: combination of Term Pair Frequency (tpf) and Inverse Document Frequency). However the results are very noisy, and while the approach maintains domain- and language-independence, these early experiments were performed at the word level, which should cause severe difficulties with morphologically rich languages. More advanced cohesion measures are currently envisaged.

Personal Publications

[1] Helena Ahonen-Myka and Antoine Doucet, "Data Mining Meets Collocations Discovery", in "Inquiries into Words, Constraints and Contexts, Festschrift for Kimmo Koskenniemi", CSLI Studies in Computational Linguistics. CSLI Publications, Center for the Study of Language and Information, University of Stanford, editors: Antti Arppe, Lauri Carlson, Krister Lindén, Jussi Piitulainen, Mickael Suominen, Martti Vainio, Hanna Westerlund and Anssi Yli-Jyrä, chapitre 19, p. 194-203, 2005.

[2] Antoine Doucet, Gabriella Kazai, Bodin Dresevic, Aleksandar Uzelac, Bogdan Radakovic and Nikola Todic, "Setting up a Competition Framework for the Evaluation of Structure Extraction from OCR-ed Books" in International Journal of Document Analysis and Recognition (IJDAR), special issue on "Performance Evaluation of Document Analysis and Recognition Algorithms", 14 (1), pp. 45-66, 2011.

[3] Gaël Dias, Rumen Moraliyski, João Cordeiro, Antoine Doucet and Helena Ahonen-Myka, "Automatic Discovery of Word Semantic Relations using Paraphrase Alignment and Distributional Lexical Semantics Analysis" in International Journal of Natural Language Engineering (JNLE), special issue on "Distributional Lexical Semantics", Cambridge Journals, 16 (4): pp. 439-467, 2010.

[4] Antoine Doucet and Helena Ahonen-Myka, "An efficient any language approach for the integration of phrases in document retrieval" in proceedings of the International Journal of Language Resources and Evaluation, special issue on "Multiword expressions: hard going or plain sailing?", Springer, 44 (1-2): p.159-180, 2010.

[5] David Alexander, Paavo Arvola, Thomas Beckers, Patrice Bellot, Timothy Chappell, C. M. DeVries, Antoine Doucet, Norbert Fuhr, Shlomo Geva, Jaap Kamps, Gabriella Kazai, Marijn Koolen, Sangeetha Kutty, Monica Landoni, Vénique Moriceau, Richi Nayak, Ragnar Nordlie, Nils Pharo, Eric SanJuan, Ralf Schenkel, Andrea Tagarelli, Xavier Tannier, James A. Thom, Andrew Trotman, Johanna Vainio, Qiuyue Wang, Chen Wu, "Report on INEX 2010", in ACM SIGIR Forum, 45 (1): p. 2-17, 2011.

[6] Thomas Beckers, Patrice Bellot, Gianluca Demartini, Ludovic Denoyer, Christopher M. De Vries, Antoine Doucet, Khairun Nisa Fachry, Norbert Fuhr, Patrick Gallinari, Shlomo Geva, Wei-Che Huang, Tereza Iofciu, Jaap Kamps, Gabriella Kazai, Marijn Koolen, Sangeetha Kutty, Monica Landoni, Miro Lehtonen, Véronique Moriceau, Richi Nayak, Ragnar Nordlie, Nils Pharo, Eric SanJuan,

Ralf Schenkel, Xavier Tannier, Martin Theobald, James A. Thom, Andrew Trotman, and Arjen P. de Vries, "Report on INEX 2009", in ACM SIGIR Forum, 44 (1): p. 38-56, 2010.

[7] Gianluca Demartini and Ludovic Denoyer and Antoine Doucet and Khairun Nisa Fachry and Patrick Gallinari and Shlomo Geva and Wei-Che Huang and Tereza Iofciu and Jaap Kamps and Gabriella Kazai and Marijn Koolen and Monica Landoni and Ragnar Nordlie and Nils Pharo and Ralf Schenkel and Martin Theobald and Andrew Trotman and Arjen P. de Vries and Alan Woodley and Jianhan Zhu, "Report on INEX 2008", in ACM SIGIR Forum, 43 (1): 20 pages, 2009.

[8] Gabriella Kazai and Antoine Doucet, "Overview of the INEX 2007 Book Search Track (BookSearch'07)", in ACM SIGIR Forum [a short version was published in INEX 2007] , 42 (1), p. 2-15, June 2008.

[9] Antoine Doucet and Helena Ahonen-Myka, "Probability and Expected Document Frequency of Discontinued Word Sequences, an Efficient Method for their Exact Computation", Traitement Automatique des Langues (TAL), "Scaling of Natural Language Processing: Complexity, Algorithms and Architectures", 46 (2): p. 13-37, 2006.

[10] Antoine Doucet, Gabriella Kazai, Jean-Luc Meunier, "ICDAR 2011 Book Structure Extraction Competition", in Proceedings of the Eleventh International Conference on Document Analysis and Recognition (ICDAR'2011), Beijing, Chine, September 18-21, p.1501-1505, 2011.

[11] Gaël Lejeune, Nadine Lucas and Antoine Doucet, " Tentative d'approche multilingue en extraction d'information. ", in Proceeding of the 10th International Conference on the Statistical Analysis of Textual Data (JADT 2010), Rome, Italy, p.1259-1268, 2010.

[12] Antoine Doucet, Helena Ahonen-Myka, "Statistical Methods for the Evaluation of Indexing Phrases", in Proceedings of the International Conference on Knowledge Discovery and Information Retrieval (KDIR 2010), Valencia, Spain, October 24-28, p. 141-149, 2010.

[13] Antoine Doucet, Gabriella Kazai, Bodin Dresevic, Aleksandar Uzelac, Bogdan Radakovic and Nikola Todic, "ICDAR 2009 Book Structure Extraction Competition", in Proceedings of the Tenth International Conference on Document Analysis and Recognition (ICDAR'2009), Barcelona, Spain, July 26-29, p.1408-1412, 2009.

[14] Miro Lehtonen and Antoine Doucet, "XML-Aided Phrase Indexing for Hypertext Documents" in Proceedings of the 31st Annual International ACM SIGIR Conference on Research and Development in Information Retrieval, Singapore, July 20-24, p.843-844, 2008.

[15] Gabriella Kazai, Antoine Doucet and Monica Landoni, "New User Tasks on Collections of Digitized Books" in Proceedings of Research and Advanced Technology for Digital Libraries, 12th European Conference, ECDL 2008, Aarhus, Denmark, September 14-19, p. 410-412, 2008.

[16] Antoine Doucet and Helena Ahonen-Myka, "Fast extraction of discontiguous sequences in text: a new approach based on maximal frequent sequences" in Proceedings of IS-LTC 2006, Information Society - Language Technologies Conference, Ljubljana, Slovenia, October 9-14, 2006, p. 186-191.

[17] Antoine Doucet, Utilisation de Séquences Fréquentes Maximales en Recherche d'Information in Proceedings of the 7th International Conference on the Statistical Analysis of Textual Data (JADT 2004), Louvain-la-Neuve, Belgium, March 10-12, 2004, p. 334-345.

[18] Gabriella Kazai, Marijn Koolen, Jaap Kamps, Antoine Doucet and Monica Landoni "Overview of the INEX 2010 Book Track: Scaling up the Evaluation using Crowdsourcing", in Advances in Focused Retrieval: 9th International Workshop of the Initiative for the Evaluation of XML Retrieval, INEX 2010, Springer LNCS, Volume Number 6932, p. 98-117, 2011.

[19] Gaël Lejeune, Antoine Doucet, Roman Yangarber and Nadine Lucas, " Filtering news for epidemic surveillance: towards processing more languages with fewer resources ", in COLING 2010, Fourth International Workshop On Cross Lingual Information Access, Beijing, China, August 2010.

[20] Gabriella Kazai, Antoine Doucet, Marijn Koolen and Monica Landoni "Overview of the INEX 2009 Book Track", in Advances in Focused Retrieval: 8th International Workshop of the Initiative for the Evaluation of XML Retrieval, INEX 2009, Springer LNCS, Volume Number 6203, p. 145-159, 2010.

[21] Gaël Lejeune, Mohammed Hatmi, Antoine Doucet, Silja Huttunen and Nadine Lucas, " A proposal for a multilingual epidemic surveillance system. ", in 1st International ICST Conference on User Centric Media (UCMedia 2009), workshop on Mining User-Generated Content for Security Workshop (Minucs 2009), Venice, Italy, 9-11 December 2009, p. 343-348, LNICST 40, 2010.

[22] Gabriella Kazai, Antoine Doucet and Monica Landoni "Overview of the INEX 2008 Book Track", in Advances in Focused Retrieval: 7th International Workshop of the Initiative for the Evaluation of XML Retrieval, INEX 2008, Springer LNCS, Volume Number 5613, p.106-123, 2009.

[23] Miro Lehtonen and Antoine Doucet, "Enhancing Keyword Search with a Keyphrase Index", in Advances in Focused Retrieval: 7th International Workshop of the Initiative for the Evaluation of XML Retrieval, INEX 2008, in Springer LNCS, Volume Number 5613, p.65-70, 2009.

[24] Antoine Doucet and Miro Lehtonen, "Let's Phrase It: INEX Topics Need Keyphrases", in ACM SIGIR 2008 Workshop on Focused Retrieval (Question Answering, Passage Retrieval, Element Retrieval), Singapore, July 20-24, p. 9-14, 2008.

[25] Gabriella Kazai and Antoine Doucet. "Overview of the INEX 2007 Book Search Track (BookSearch'07)", in Focused access to XML documents, Sixth International Workshop of the Initiative for the Evaluation of XML Retrieval, INEX 2007, Springer LNCS, Volume Number 4862, p. 148-161, 2008.

[26] Miro Lehtonen and Antoine Doucet, "Phrase detection in the Wikipedia", in Focused access to XML documents, Sixth International Workshop of the Initiative for the Evaluation of XML Retrieval, INEX 2007, Springer LNCS, Volume Number 4862, p. 115-121, 2008.

[27] Antoine Doucet and Miro Lehtonen, "Unsupervised classification of text-centric XML document collections", in Comparative Evaluation of XML Information Retrieval Systems, Fifth International Workshop of the Initiative for the Evaluation of XML Retrieval, INEX 2006, Springer LNCS, Volume Number 4518, p.497-509, 2007.

[28] Miro Lehtonen and Antoine Doucet, "EXTIRP: baseline retrieval from Wikipedia", in Comparative Evaluation of XML Information Retrieval Systems, Fifth International Workshop of the Initiative for the Evaluation of XML Retrieval, INEX 2006, Springer LNCS, Volume Number 4518, p.119-124, 2007.

[29] Antoine Doucet and Helena Ahonen-Myka, A Method to Calculate Probability and Expected Document Frequency of Discontinued Word Sequences in Proceedings of ACM SIGIR 2005, ELECTRA Workshop on Methodologies and Evaluation of Lexical Cohesion Techniques in Real-world Applications (Beyond Bag of Words), Salvador, Brazil, August 15-19, 2005, p. 33-40.

[30] Antoine Doucet and Helena Ahonen-Myka, Non-Contiguous Word Sequences for Information Retrieval in Proceedings of the 42nd annual meeting of the Association for Computational Linguistics (ACL-2004), Workshop on Multiword Expressions: Integrating Processing, Barcelona, Spain, July 21-26, 2004, p. 88-95.

[31] Antoine Doucet, Lili Aunimo, Miro Lehtonen and Renaud Petit, Accurate Retrieval of XML Document Fragments using EXTIRP in Proceedings of the Second Annual Workshop of the Initiative for the Evaluation of XML retrieval (INEX), Schloss Dagstuhl, Germany, December 15-17, 2003, ERCIM Workshop Proceedings, 2004, 8 pages.

[32] Antoine Doucet and Helena Ahonen-Myka, Naive clustering of a large XML document collection in Proceedings of the First Annual Workshop of the Initiative for the Evaluation of XML retrieval (INEX), Schloss Dagstuhl, Germany, December 9-11, 2002, ERCIM Workshop Proceedings, March 2003, p. 81-88.

[33] Thierry Charnois, Antoine Doucet, Yann Mathet and François Rioult, "3 approches du GREYC pour la classification de textes" in Proceedings of DEfi Fouille de Texte (DEFT'08), Avignon, France, p. 171-180, 2008.

[34] Antoine Doucet, Extracting More Relevant Document Descriptors using Hierarchical Information in Proceedings of XML Finland 2002, October 21-22, p. 136-147.

[35] Antoine Doucet, Améliorer les descripteurs de documents semi-structurés en utilisant les informations contextuelles. INFORSID 2002, Forum Jeunes Chercheurs, Nantes, France, June 4-7, 2002, p. 401-402.

[36] Antoine Doucet, "Advanced Document Description. A Sequential Approach", ISBN 952-10-2801-7, Helsinki University Printing House, 161 pages, November 2005.

[37] Antoine Doucet, "Advanced Document Descriptors, a Sequential Approach", Dissertation Abstract, SIGIR Forum 40 (1): p.71-72, 2006.

[38] Antoine Doucet, "Opponentti, Kustos, Karonkka, jne.", en finnois dans la revue "*Yliopistolainen*" (l'Universitaire) 125 (2), mensuel publié en ligne et à 10 000 exemplaires papier, Helsinki University Printing House, février 2007, p.10.

[39] Antoine Doucet, "Prendre les mots dans le bon sens : une question d'ordre", dans la revue "*Universitas Helsingiensis*" 44 (4), trimestriel multilingue publié en ligne et à 10 000 exemplaires papier, Helsinki University Printing House, Décembre 2006, p.36-38.

Bibliography

[AD04] Luis Von Ahn and Laura Dabbish. Labeling images with a computer game. In *Proceedings of the SIGCHI conference on Human factors in computing systems*, CHI '04, pages 319–326, New York, NY, USA, 2004. ACM.

[All02] James Allan. *Introduction to topic detection and tracking*, pages 1–16. Kluwer Academic Publishers, Norwell, MA, USA, 2002.

[AS95] Rakesh Agrawal and Ramakrishnan Srikant. Mining sequential patterns. In Yu and Chen, editors, *11th International Conference on Data Engineering*, pages 3–14, Taipei, Taiwan, 1995. IEEE Computer Society Press.

[AVdG09] Martin Atkinson and Erik Van der Goot. Near real time information mining in multilingual news. In *Proceedings of the 18th international conference on World wide web*, WWW '09, pages 1153–1154, New York, NY, USA, 2009. ACM.

[BACG+03] Yiftah Ben-Aharon, Sara Cohen, Yael Grumbach, Yaron Kanza, Jonathan Mamou, Yehoshua Sagiv, Benjamin Sznajder, and Efrat Twito. Searching in an XML Corpus Using Content and Structure. In *Proceedings of the Second Workshop of the Initiative for the Evaluation of XML Retrieval (INEX)*, Schloss Dagsuhl, Germany, 2003.

[BL03] Regina Barzilay and Lillian Lee. Learning to paraphrase: an unsupervised approach using multiple-sequence alignment. In *Proceedings of the 2003 Conference of the North American Chapter of the Association for Computational Linguistics on Human Language Technology - Volume 1*, NAACL '03, pages 16–23, Stroudsburg, PA, USA, 2003. Association for Computational Linguistics.

[BPSM+04] Tim Bray, Jean Paoli, C. M. Sperberg-McQueen, Eve Maler, and Francois Yergeau. Extensible Markup Language (XML) 1.0 - W3C recommendation 04 february 2004. Technical Report REC-xml-20040204, 2004.

[BPSY08] Sivaji Bandyopadhyay, Thierry Poibeau, Horacio Saggion, and Roman Yangarber, editors. *Coling 2008: Proceedings of the workshop Multi-source Multilingual Information Extraction and Summarization.* Coling 2008 Organizing Committee, Manchester, UK, August 2008.

[BSB10] Plaban Kumar Bhowmick, Sudeshna Sarkar, and Anupam Basu. Ontology based user modeling for personalized information access. *IJCSA*, 7(1):1–22, 2010.

[BYRN99] R. Baeza-Yates and B. Ribeiro-Neto. *Modern Information Retrieval*. Addison-Wesley, ACM Press, New York, 1999.

[CCSC10] Charlies Clarke, Nick Craswell, Ian Soboroff, and Gordon Cormack. Overview of the TREC-2010 Web Track. In *Proceedings of TREC-2010*, Gaithersburg, Maryland USA, November 2010.

[CCSV11] Charlies Clarke, Nick Craswell, Ian Soboroff, and Ellen Voorhees. Overview of the TREC-2011 Web Track. In *Proceedings of TREC-2011*, Gaithersburg, Maryland USA, November 2011.

[CDB07] João Cordeiro, Gaël Dias, and Pavel Brazdil. New functions for unsupervised asymmetrical paraphrase detection. *JSW*, 2(4):12–23, 2007.

[CH90] Kenneth W. Church and Patrick Hanks. Word association norms, mutual information, and lexicography. *Computational Linguistics*, 16(1):22–29, 1990.

[CH03] Nick Craswell and David Hawking. Overview of the TREC-2003 Web Track. In *Proceedings of TREC-2003*, Gaithersburg, Maryland USA, November 2003.

[CH04] Nick Craswell and David Hawking. Overview of the TREC-2004 Web Track. In *Proceedings of TREC-2004*, Gaithersburg, Maryland USA, November 2004.

[CKJ+06] Nigel Collier, Ai Kawazoe, Lihua Jin, Mika Shigematsu, Dinh Dien, Roberto Barrero, Koichi Takeuchi, and Asanee Kawtrakul. A multilingual ontology for infectious disease surveillance: rationale, design and challenges. *Language Resources and Evaluation*, 40:405–413, 2006. 10.1007/s10579-007-9019-7.

[CKN83] Yaacov Choueka, Shmuel T. Klein, and E. Neuwitz. Automatic retrieval of frequent idiomatic and collocational expressions in a large corpus. *Journal for Literary and Linguistic computing*, 4:34–38, 1983.

[CL96] Jim Cowie and Wendy Lehnert. Information extraction. *Commun. ACM*, 39:80–91, January 1996.

[Cla05] Charles L. A. Clarke. Controlling overlap in content-oriented xml retrieval. In *SIGIR '05: Proceedings of the 28th annual international ACM SIGIR conference on Research and development in information retrieval*, pages 314–321, New York, NY, USA, 2005. ACM Press.

[CLH11] Xiaofeng Zhang Jie Liu Caihua Liu, Jiajun Chen and Yalou Huang. Toc structure extraction from ocr-ed books. In Shlomo Geva, Jaap

Kamps, and Andrew Trotman, editors, *Focused Retrieval and Evaluation*, Lecture Notes in Computer Science, pages 70–80. Springer Berlin / Heidelberg, 2011.

[Coy06] K. Coyle. Mass digitization of books. *Journal of Academic Librarianship*, 32(6):641–645, 2006.

[CP03] Roger Chartier and Alain Paire. *Pratiques de la lecture / sous la direction de Roger Chartier et à l'initiative d'Alain Paire.* Payot & Rivages, Paris, 2003. Publication originale chez Rivages en 1985.

[Cro05] Bruce Croft. Keynote speech: Phrases and other structure in queries. In *SIGIR '05: Proceedings of the 28th annual international ACM SIGIR conference on Research and development in information retrieval. Workshop on Methodologies and Evaluation of Lexical Cohesion Techniques in Real-World Applications (ELECTRA)*, New York, NY, USA, 2005. ACM Press.

[CSW97] Michal Cutler, Yungming Shih, and Meng Weiyi. Using the structure of html documents to improve retrieval. In *Proceedings of the USENIX Symposium on Internet Technologies and Systems (NISTS'97)*, 1997.

[CSYT05] David Carmel, Ian Soboroff, and Elad Yom-Tov, editors. *Workshop on Predicting Query Difficulty - Methods and Applications*, 2005.

[CTT05] L. Candillier, I. Tellier, and F. Torre. Transforming xml trees for efficient classification and clustering. INEX 2005 Workshop on Mining XML documents, nov. 2005.

[CW87] Don Coppersmith and Shmuel Winograd. Matrix multiplication via arithmetic progressions. In *STOC'87: Proceedings of the 19th annual ACM conference on Theory of computing*, pages 1–6, 1987.

[CY92] C. J. Crouch and B. Yang. Experiments in automatic statistical thesaurus construction. In *Proceedings of the 15th ACM SIGIR Conference on Research and Development in Information Retrieval*, pages 77–88, Copenhagen, Denmark, 1992.

[CYTDP06] David Carmel, Elad Yom-Tov, Adam Darlow, and Dan Pelleg. What makes a query difficult? In Efthimis N. Efthimiadis, Susan T. Dumais, David Hawking, and Kalervo Järvelin, editors, *SIGIR 2006: Proceedings of the 29th Annual International ACM SIGIR Conference on Research and Development in Information Retrieval, Seattle, Washington, USA, August 6-11, 2006*, pages 390–397. ACM, 2006.

[DG06] Ludovic Denoyer and Patrick Gallinari. The Wikipedia XML Corpus. *SIGIR Forum*, 2006.

[DG07] Ludovic Denoyer and Patrick Gallinari. Report on the xml mining track at inex 2005 and inex 2006. In Fuhr et al. [FLMK07].

[DGBPL00] Gaël Dias, Sylvie Guilloré, Jean-Claude Bassano, and José Gabriel Pereira Lopes. Extraction automatique d'unités complexes: Un enjeu fondamental pour la recherche documentaire. *Traitement Automatique des Langues*, 41(2):447–472, 2000.

[Dia03] Gaël Dias. Multiword unit hybrid extraction. In *Workshop on Multiword Expressions of the 41st ACL meeting. Sapporo. Japan.*, 2003.

[DLTV05] Thierry Despeyroux, Yves Lechevallier, Brigitte Trousse, and Anne-Marie Vercoustre. Experiments in clustering homogeneous xml documents to validate an existing typology. In *Proceedings of the 5th International Conference on Knowledge Management (I-Know)*, Vienna, Austria, July 2005. Journal of Universal Computer Science.

[DM09] Hervé Déjean and Jean-Luc Meunier. Xrce participation to the book structure task. In Shlomo Geva, Jaap Kamps, and Andrew Trotman, editors, *Advances in Focused Retrieval*, volume 5631 of *Lecture Notes in Computer Science*, pages 124–131. Springer Berlin / Heidelberg, 2009.

[DM10a] Hervé Déjean and Jean-Luc Meunier. Xrce participation to the 2009 book structure task. In Shlomo Geva, Jaap Kamps, and Andrew Trotman, editors, *Advances in Focused Retrieval: 8th International Workshop of the Initiative for the Evaluation of XML Retrieval (INEX 2009)*, volume 6203 of *Lecture Notes in Computer Science*, pages 160–169. Springer Berlin / Heidelberg, 2010.

[DM10b] Hervé Déjean and Jean-Luc Meunier. Xrce participation to the 2009 book structure task. In Shlomo Geva, Jaap Kamps, and Andrew Trotman, editors, *Focused Retrieval and Evaluation*, volume 6203 of *Lecture Notes in Computer Science*, pages 160–169. Springer Berlin / Heidelberg, 2010.

[Dor03] Daniel Dor. On newspaper headlines as relevance optimizers. *Journal of Pragmatics*, 35(5):695–721, May 2003.

[DTB11] Mariam Daoud, Lynda Tamine, and Mohand Boughanem. A personalized search using a semantic distance measure in a graph-based ranking model. *J. Information Science*, 37(6):614–636, 2011.

[DURT09] Bodin Dresevic, Aleksandar Uzelac, Bogdan Radakovic, and Nikola Todic. Book layout analysis: Toc structure extraction engine. In Shlomo Geva, Jaap Kamps, and Andrew Trotman, editors, *Advances in Focused Retrieval*, volume 5631 of *Lecture Notes in Computer Science*, pages 164–171. Springer Berlin / Heidelberg, 2009.

[EBSW08] Oren Etzioni, Michele Banko, Stephen Soderland, and Daniel S. Weld. Open information extraction from the web. *Commun. ACM*, 51:68–74, December 2008.

[EFC+11] Oren Etzioni, Anthony Fader, Janara Christensen, Stephen Soderland, and Mausam. Open information extraction: The second generation. In *IJCAI*, pages 3–10, 2011.

[Fag89] Joel L. Fagan. The effectiveness of a nonsyntactic approach to automatic phrase indexing for document retrieval. *Journal of the American Society for Information Science*, 40:115–132, 1989.

[Fan61] Robert M. Fano. *Transmission of Information: A Statistical Theory of Information*. MIT Press, Cambridge MA, 1961.

[FAT98] Katerina Frantzi, Sophia Ananiadou, and Jun-ichi Tsujii. The c-value/nc-value method of automatic recognition for multi-word terms. In *ECDL '98: Proceedings of the Second European Conference on Research and Advanced Technology for Digital Libraries*, pages 585–604. Springer-Verlag, 1998.

[Fel68] William Feller. *An Introduction to Probability Theory and Its Applications*, volume 1. Wiley Publications, third edition, 1968.

[FGKL02] Norbert Fuhr, Norbert Gövert, Gabriella Kazai, and Mounia Lalmas, editors. *Proceedings of the First Workshop of the INitiative for the Evaluation of XML Retrieval (INEX), Schloss Dagstuhl, Germany, December 9-11, 2002*, 2002.

[FKLT08] Norbert Fuhr, Jaap Kamps, Mounia Lalmas, and Andrew Trotman, editors. *Focused Access to XML Documents, 6th International Workshop of the Initiative for the Evaluation of XML Retrieval, INEX 2007, Dagstuhl Castle, Germany, December 17-19, 2007. Selected Papers*, volume 4862 of *Lecture Notes in Computer Science*. Springer, 2008.

[FLMK06] Norbert Fuhr, Mounia Lalmas, Saadia Malik, and Gabriella Kazai, editors. *Advances in XML Information Retrieval and Evaluation, 4th International Workshop of the Initiative for the Evaluation of XML Retrieval, INEX 2005, Dagstuhl Castle, Germany, November 2005, Revised Selected Papers*, volume 3977 of *Lecture Notes in Computer Science*. Springer, 2006.

[FLMK07] Norbert Fuhr, Mounia Lalmas, Saadia Malik, and Gabriella Kazai, editors. *Advances in XML Information Retrieval and Evaluation, 5th International Workshop of the Initiative for the Evaluation of XML Retrieval, INEX 2006, Dagstuhl Castle, Germany, December 18-20 2006, Revised Selected Papers*, Lecture Notes in Computer Science. Springer, 2007.

[Fox83] Edward A. Fox. Some considerations for implementing the smart information retrieval system under unix. Technical Report TR 83-560, Department of Computer Science, Cornell University, Ithaca, NY, September 1983.

[GKK09] Fredric Gey, Jussi Karlgren, and Noriko Kando. Information access in a multilingual world: transitioning from research to real-world applications. *SIGIR Forum*, 43:24–28, December 2009.

[GKST11] Shlomo Geva, Jaap Kamps, Ralf Schenkel, and Andrew Trotman, editors. *Comparative Evaluation of Focused Retrieval - 9th International Workshop of the Inititative for the Evaluation of XML Retrieval, INEX 2010, Vugh, The Netherlands, December 13-15, 2010, Revised Selected Papers*, volume 6932 of *Lecture Notes in Computer Science*. Springer, 2011.

[GKT09] Shlomo Geva, Jaap Kamps, and Andrew Trotman, editors. *Advances in Focused Retrieval, 7th International Workshop of the Initiative for the Evaluation of XML Retrieval, INEX 2008, Dagstuhl Castle, Germany, December 15-18, 2008. Revised and Selected Papers*, volume 5631 of *Lecture Notes in Computer Science*. Springer, 2009.

[GKT10] Shlomo Geva, Jaap Kamps, and Andrew Trotman, editors. *Focused Retrieval and Evaluation, 8th International Workshop of the Initiative for the Evaluation of XML Retrieval, INEX 2009, Brisbane, Australia, December 7-9, 2009, Revised and Selected Papers*, volume 6203 of *Lecture Notes in Computer Science*. Springer, 2010.

[GL10] Emmanuel Giguet and Nadine Lucas. The book structure extraction competition with the resurgence software at caen university. In Shlomo Geva, Jaap Kamps, and Andrew Trotman, editors, *Focused Retrieval and Evaluation*, volume 6203 of *Lecture Notes in Computer Science*, pages 170–178. Springer Berlin / Heidelberg, 2010.

[GM00] Damien Guillaume and Fionn Murtaugh. Clustering of XML Documents. *Computer Physics Communications*, 127:215–227, 2000.

[Hav03] Taher H. Haveliwala. Topic-sensitive pagerank: A context-sensitive ranking algorithm for web search. *IEEE Transactions on Knowledge and Data Engineering*, 15(4):784–796, 2003.

[HJ94] Roger A. Horn and Charles R. Johnson. *Topics in matrix analysis*. Cambridge University Press, New York, NY, USA, 1994.

[Hob93] Jerry R. Hobbs. The generic information extraction system. In *Proceedings of the 5th conference on Message understanding*, MUC 5 '93, pages 87–91, Stroudsburg, PA, USA, 1993. Association for Computational Linguistics.

[IS10] Hazra Imran and Aditi Sharan. Co-occurrence based predictors for estimating query difficulty. In *Proceedings of the 2010 IEEE International Conference on Data Mining Workshops*, ICDMW '10, pages 867–874, Washington, DC, USA, 2010. IEEE Computer Society.

[JS74] Stuart Jones and John Mc Hardy Sinclair. English lexical collocations: A study in computational linguistics. *Cahiers de Lexicologie*, 24:15–61, 1974.

[JvR71] N. Jardine and C. J. van Rijsbergen. The use of hierarchic clustering in information retrieval. *Information Storage and Retrieval*, 7:217–240, 1971.

[KFB09] Mikaela Keller, Clark C. Freifeld, and John S. Brownstein. Automated vocabulary discovery for geo-parsing online epidemic intelligence. *BMC Bioinformatics*, 10:385, 2009.

[KGT08] Jaap Kamps, Shlomo Geva, and Andrew Trotman. Report on the SIGIR 2008 workshop on focused retrieval. *SIGIR Forum*, 42(2):59–65, 2008.

[KHT+07] M. Kc, M. Hagenbuchner, A. C. Tsoi, F. Scarselli, M. Gori, and A. Sperduti. Xml document mining using contextual self-organizing maps for structures. In Fuhr et al. [FLMK07].

[KKMFW08] Paul Kantor, Gabriella Kazai, Natasa Milic-Frayling, and Ross Wilkinson, editors. *BooksOnline '08: Proceeding of the 2008 ACM Workshop on Research Advances in Large Digital Book Repositories*, New York, NY, USA, 2008. ACM.

[KL06] Gabriella Kazai and Mounia Lalmas. extended cumulated gain measures for the evaluation of content-oriented xml retrieval. *ACM Trans. Inf. Syst.*, 24:503–542, October 2006.

[KLdV04] Gabriella Kazai, Mounia Lalmas, and Arjen P. de Vries. The overlap problem in content-oriented xml retrieval evaluation. In *SIGIR '04: Proceedings of the 27th annual international ACM SIGIR conference on Research and development in information retrieval*, pages 72–79, New York, NY, USA, 2004. ACM Press.

[Kle99] Jon M. Kleinberg. Authoritative sources in a hyperlinked environment. *Journal of the ACM (JACM)*, 46(5):604–632, 1999.

[KMFC09] Gabriella Kazai, Natasa Milic-Frayling, and Jamie Costello. Towards methods for the collective gathering and quality control of relevance assessments. In *Proceedings of the 32nd international ACM SIGIR conference on Research and development in information retrieval*, SIGIR '09, pages 452–459, New York, NY, USA, 2009. ACM.

[KRV11] Valia Kordoni, Carlos Ramisch, and Aline Villavicencio, editors. *ACL Workshop on Multiword Expressions: from Parsing and Generation to the Real World (MWE 2011)*, 2011.

[KSB06] Juha Kärkkäinen, Peter Sanders, and Stefan Burkhardt. Linear work suffix array construction. *Journal of the ACM*, 53:918–936, November 2006.

[Lam96] Knud Lambrecht. *Information Structure and Sentence Form: Topic, Focus, and the Mental Representations of Discourse Referents (Cambridge Studies in Linguistics)*. Cambridge University Press, November 1996.

[Lee95] Joon Ho Lee. Combining multiple evidence from different properties of weighting schemes. In *SIGIR '95: Proceedings of the 18th annual international ACM SIGIR conference on Research and development in information retrieval*, pages 180–188. ACM Press, 1995.

[Leh05] Miro Lehtonen. EXTIRP 2004: Towards heterogeneity. In Norbert Fuhr, Mounia Lalmas, Saadia Malik, and Zoltán Szlávik, editors, *INEX*, volume 3493 of *Lecture Notes in Computer Science*, pages 372–381. Springer, 2005.

[Leh06a] Miro Lehtonen. *Indexing Heterogeneous XML for Full-Text Search.* PhD thesis, University of Helsinki, 2006.

[Leh06b] Miro Lehtonen. Preparing Heterogeneous XML for Full-Text Search. *ACM Transactions on Information Systems*, 24(4):1–21, October 2006.

[Leh06c] Miro Lehtonen. When a few highly relevant answers are enough. In Fuhr et al. [FLMK06].

[Leh07] Miro Lehtonen. Vocabulary-independent methods for xml information retrieval. In *Proceedings of Workshop on Advances in Methods of Information and Communication Technology (AMICT 06)*, pages 53–61, 2007.

[Lem03] Lemur. Lemur Toolkit for Language Modeling and IR, 2003.

[Lev66] V. I. Levenshtein. Binary codes capable of correcting deletions, insertions and reversals. *Soviet Physics Doklady*, 10:707+, February 1966.

[Lew92] David Dolan Lewis. *Representation and learning in information retrieval.* PhD thesis, University of Massachusetts at Amherst, 1992.

[LT07] Mounia Lalmas and Anastasios Tombros. Inex 2002 - 2006: Understanding xml retrieval evaluation. In Costantino Thanos, Francesca Borri, and Leonardo Candela, editors, *Digital Libraries: Research and Development*, volume 4877 of *Lecture Notes in Computer Science*, pages 187–196. Springer Berlin / Heidelberg, 2007.

[Luc12] Nadine Lucas. *Stylistic devices in the news, as related to topic recognition*, pages 301–316. Peter Lang, 2012.

[LZ03] Dekang Lin and Shaojun Zhao. Identifying synonyms among distributionally similar words. In *In Proceedings of IJCAI-03*, pages 1492–1493, 2003.

[MBSC87] Mandar Mitra, Chris Buckley, Amit Singhal, and Claire Cardie. An analysis of statistical and syntactic phrases. In *Proceedings of RIAO97, Computer-Assisted Information Searching on the Internet*, pages 200–214, 1987.

[MBSC97] Mandar Mitra, Chris Buckley, Amit Singhal, and Claire Cardie. An analysis of statistical and syntactic phrases. In *Proceedings of RIAO97, Computer-Assisted Information Searching on the Internet*, pages 200–214, 1997.

[McC06] Andrew McCallum. Information extraction, data mining and joint inference. In *KDD*, page 835, 2006.

[MS99] Christopher D. Manning and Hinrich Schütze. *Foundations of Statistical Natural Language Processing*. MIT Press, Cambridge MA, second edition, 1999.

[muc91] *Proceedings of the 3rd Conference on Message Understanding, MUC 1991, San Diego, California, USA, May 21-23, 1991*. ACL, 1991.

[muc92] *Proceedings of the 4th Conference on Message Understanding, MUC 1992, McLean, Virginia, USA, June 16-18, 1992*, 1992.

[Nó4] Claire Nédellec. Machine learning for information extraction in genomics - state of the art and perspectives, text mining and its applications. In *Results of the NEMIS Launch Conference Series: Studies in Fuzziness and Soft Computing, Sirmakessis, Spiros (Ed.), Springer Verlag. Nédellec C., Ould Abdel Vetah M. and Bessières P*, pages 99–118. Springer-Verlag, 2004.

[ND77] Ben Noble and James W. Daniel. *Applied Linear Algebra*, pages 361–367. Prentice Hall, second edition, 1977.

[NJ02] Andrew Nierman and H.V. Jagadish. Evaluating Structural Similarity in XML. In *Fifth International Workshop on the Web and Databases (WebDB 2002), Madison, Wisconsin*, 2002.

[Not07] Cédric Notredame. Recent Evolutions of Multiple Sequence Alignment Algorithms. *PLoS Computational Biology*, 3(8):123, 2007.

[PBMW98] Lawrence Page, Sergey Brin, Rajeev Motwani, and Terry Winograd. The pagerank citation ranking: Bringing order to the web. Technical report, Stanford Digital Library Technologies Project, 1998.

[QF93] Y Qiu and H Frei. Concept based query expansion. In *Proceedings of the 16th ACM SIGIR Conference on Research and Development in Information Retrieval*, pages 160–169, Pittsburgh, PA, USA, 1993.

[QLZ+05] Tao Qin, Tie-Yan Liu, Xu-Dong Zhang, Zheng Chen, and Wei-Ying Ma. A study of relevance propagation for web search. In *SIGIR '05: Proceedings of the 28th annual international ACM SIGIR conference on Research and development in information retrieval*, pages 408–415, New York, NY, USA, 2005. ACM Press.

[Reu87] Reuters-21578. Text categorization test collection, distribution 1.0, 1987. http://www.daviddlewis.com/resources/testcollections/reuters21578.

[Rij79] C. J. Van Rijsbergen. *Information retrieval*. Butterworths, London, 2 edition, 1979.

[RLHJ99] Dave Raggett, Arnaud Le Hors, and Ian Jacobs. Html 4.01 specification - W3C recommendation 24 december 1999. Technical Report REC-html401-19991224, 1999.

[Roc71] J. J. Rocchio. *Relevance feedback in information retrieval. In Salton, G., editor, The SMART Retrieval System - Experiments in Automatic Document Processing*. Prentice Hall Inc., 1971.

[RZT03] Stephen E. Robertson, Hugo Zaragoza, and Michael Taylor. Microsoft cambridge at trec-12: Hard track. In *TREC*, pages 418–425, 2003.

[SB88] Gerard Salton and Chris Buckley. Term-weighting approaches in automatic text retrieval. *Information Processing and Management: an International Journal*, 24(5):513–523, 1988.

[SC96] Tomek Strzalkowski and Jose Perez Carballo. Natural language information retrieval: TREC-4 report. In *Text REtrieval Conference*, pages 245–258, 1996.

[Seb02] Fabrizio Sebastiani. Machine learning in automated text categorization. *ACM Computing Survey*, 34(1):1–47, 2002.

[Sen11] Gianluca Sensidoni. Who, what, when, where and how: Semantics semantics help connect the dots. In *European Intelligence and Security Informatics Conference, EISIC 2011, Athens, Greece, September 12-14, 2011*, page 9, 2011.

[SFvdG+08] Ralf STEINBERGER, Flavio FUART, Erik van der GOOT, Clive BEST, Peter von ETTER, and Roman YANGARBER. Text mining from the web for medical intelligence. 2008.

[SHY11] Peter von Etter Silja Huttunen, Arto Vihavainen and Roman Yangarber. Relevance prediction in information extraction using discourse and lexical features. In *Proceedings of the 18th International Nordic Conference of Computational Linguistics (NODALIDA-2011)*. NEALT, 2011.

[Sin97] Amitabh K. Singhal. *Term Weighting Revisited*. PhD thesis, Cornell University, 1997.

[SJVR75] Karen Spärck Jones and Keith Van Rijsbergen. Report on the need for and provision of an "ideal" information retrieval test collection. British Library Research and Development Report 5266, Computer Laboratory, University of Cambridge, 1975.

[Sma93] Frank Smadja. Retrieving collocations from text: Xtract. *Journal of Computational Linguistics*, 19:143–177, 1993.

[Ste11] Ralf Steinberger. A survey of methods to ease the development of highly multilingual text mining applications. *Language Resources and Evaluation*, pages 1–22, 2011.

[STF11] Djamé Seddah, Reut Tsarfaty, and Jennifer Foster, editors. *Proceedings of the Second Workshop on Statistical Parsing of Morphologically Rich Languages*. Association for Computational Linguistics, Dublin, Ireland, October 2011.

[SYY75] Gerard Salton, C.S. Yang, and Clement T. Yu. A theory of term importance in automatic text analysis. *Journal of the American Society for Information Science*, 26(1):33–44, 1975.

[SZ03] Azadeh Shakery and ChengXiang Zhai. Relevance propagation for topic distillation uiuc trec 2003 web track experiments. In *Proceedings of TREC-2003*, pages 673–677, Gaithersburg, Maryland USA, November 2003.

[TG01] Ilias Tsoukatos and Dimitrios Gunopulos. Efficient mining of spatiotemporal patterns. In *Proceedings of the 7th International Symposium on Advances in Spatial and Temporal Databases (SSTD)*, pages 425–442, 2001.

[TM99] Andrew Turpin and Alistair Moffat. Statistical phrases for vector-space information retrieval. In *Proceedings of the 22nd ACM SIGIR Conference on Research and Development in Information Retrieval*, pages 309–310, 1999.

[Tom02] Anastasios Tombros. *The effectiveness of hierarchic query-based clustering of documents for information retrieval*. PhD thesis, University of Glasgow, 2002.

[Ukk09] Esko Ukkonen. Maximal and minimal representations of gapped and non-gapped motifs of a string. *Theoretical Computer Science*, 410:4341–4349, October 2009.

[Van99] Christian Vandendorpe. *Du papyrus à l'hypertexte. Essai sur les mutations du texte et de la lecture*. Boréal, Montréal, 1999. nt2 hypertexte.

[Vec05] Olga Vechtomova. The role of multi-word units in interactive information retrieval. In *Proceedings of the 27th European Conference on Information Retrieval, Santiago de Compostela, Spain*, pages 403–420, 2005.

[Ver02] Jacques Vergne. Une méthode pour l'analyse descendante et calculatoire de corpus multilingues : application au calcul des relations sujet-verbe. In *Actes de Traitement Automatique de la Langue (TALN 2002)*, pages 63–74, 2002.

[VFLD06] Anne-Marie Vercoustre, Mounir FEGAS, Yves Lechevallier, and Thierry Despeyroux. Classification de documents xml à partir d'une

représentation linéaire des arbres de ces documents. In *Actes des 6èmes journées Extraction et Gestion des Connaissances (EGC 2006), Revue des Nouvelles Technologies de l'Information (RNTI-E-6)*, Lille, France, January 2006.

[Vin07] L. Vincent. Google book search: Document understanding on a massive scale. In *ICDAR*, pages 819–823, 2007.

[Voo92] E. Voorhees. Relevance feedback revisited. In *Proceedings of the 15th ACM SIGIR Conference on Research and Development in Information Retrieval*, pages 1–10, Copenhagen, Denmark, 1992.

[Voo94] E. Voorhees. Query expansion using lexical-semantic relations. In *Proceedings of the 17th ACM SIGIR Conference on Research and Development in Information Retrieval*, pages 61–69, Dublin, Ireland, 1994.

[Wil88] Peter Willett. Recent trends in hierarchic document clustering: a critical review. *Information Processing and Management*, 24(5):577–597, 1988.

[WKT08] Hengzhi Wu, Gabriella Kazai, and Michael Taylor. Book search experiments: Investigating IR methods for the indexing and retrieval of books. In *Advances in Information Retrieval. Proceedings of the 30th European Conference on Information Retrieval*, volume 4956 of *Lecture Notes in Computer Science*, pages 234–245. Springer, 2008.

[WSC09] *WSCD '09: Proceedings of the 2009 workshop on Web Search Click Data*, New York, NY, USA, 2009. ACM.

[WZB04] Hugh E. Williams, Justin Zobel, and Dirk Bahle. Fast phrase querying with combined indexes. *ACM Transactions on Information Systems*, 22(4):573–594, 2004.

[YHT+06] S. L. Yong, M. Hagenbuchner, A. Tsoi, F. Scarselli, and M. Gori. Xml document mining using graph neural network. In Fuhr et al. [FLMK06].

[YS00] Jeonghee Yi and Neel Sundaresan. A classifier for semi-structured documents. In *Proceedings of the sixth ACM SIGKDD international conference on Knowledge discovery and data mining, Boston, Massachusetts*, pages 340–344, 2000.

[Zak01] Mohammed J. Zaki. Spade: An efficient algorithm for mining frequent sequences. *Machine Learning*, 42(1-2):31–60, 2001.

[ZL07] Roelof Van Zwol and Tim Van Loosbroek. Effective use of semantic structure in XML retrieval. In *ECIR*, pages 621–628, 2007.