Mario Walter Taschwer

Concept-Based and Multimodal Methods for Medical Case Retrieval

DOCTORAL THESIS

submitted in fulfilment of the requirements for the degree of

Doktor der technischen Wissenschaften

Alpen-Adria-Universität Klagenfurt

Fakultät für Technische Wissenschaften

Mentor
O.Univ.-Prof. Dr. Laszlo Böszörmenyi
Alpen-Adria-Universität Klagenfurt
Institut für Informationstechnologie

Evaluator
O.Univ.-Prof. Dr. Laszlo Böszörmenyi
Alpen-Adria-Universität Klagenfurt
Institut für Informationstechnologie

Evaluator
Professor Oge Marques Ph.D.
Florida Atlantic University, Boca Raton, FL, USA
Department of Computer and Electrical Engineering
and Computer Science

Klagenfurt, March 2017

Affidavit

I hereby declare in lieu of an oath that

- the submitted academic paper is entirely my own work and that no auxiliary materials have been used other than those indicated,
- I have fully disclosed all assistance received from third parties during the process of writing the paper, including any significant advice from supervisors,
- any contents taken from the works of third parties or my own works that have been included either literally or in spirit have been appropriately marked and the respective source of the information has been clearly identified with precise bibliographical references (e.g. in footnotes),
- to date, I have not submitted this paper to an examining authority either in Austria or abroad and that
- the digital version of the paper submitted for the purpose of plagiarism assessment is fully consistent with the printed version.

I am aware that a declaration contrary to the facts will have legal consequences.

(Signature) (Place, Date)

Acknowledgments

First of all, I would like to thank my supervisors Prof. Laszlo Böszörmenyi and Prof. Oge Marques for their continuous support and feedback during the four years of working on this thesis. They created an atmosphere of appreciation and friendship that helped me to sustain the efforts needed to complete this work. In particular, special thanks go to Oge Marques for countless discussions and suggestions for improvement during all stages of PhD development. I am also grateful to my colleagues at the ITEC department of AAU, who bore with me at times of reduced capacity with respect to IT administration tasks. Special gratitude is owed to Martina Steinbacher and Rudolf Messner for their support in reducing my administrative duty when required by scientific work. Last but not least, my warmest thanks go to my wife Monika and our five children Elisabeth, Sarah, Johannes, Magdalena, and Judith, with whom I could share successful as well as disappointing moments during work, and who thereby provided a social and emotional backing that helped me to pursue this PhD project over the years.

Zusammenfassung

Die Suche in medizinischen Fallbeschreibungen (Medical Case Retrieval, MCR) ist als Multimedia-Suchproblem in einer Dokumentsammlung aus Fallbeschreibungen definiert, die bestimmte Erkrankungen, Krankengeschichten von Patienten oder andere Einheiten von biomedizinischem Wissen betreffen. Fallbeschreibungen sind Multimedia-Dokumente, die Text- und Bildmodalitäten enthalten. Eine Suchanfrage kann aus einer textuellen Beschreibung der Symptome eines Patienten und damit zusammenhängenden diagnostischen Bildern bestehen. Diese Dissertation untersucht und bewertet Verfahren, die auf eine Verbesserung der Effektivität von MCR im Vergleich zur Volltextsuche abzielen. Wir vertreten die Hypothese, dass dieses Ziel durch die Ausnützung von kontrollierten Vokabularien von biomedizinischen Begriffen für die Erweiterung von Suchanfragen und für konzeptbasierte Suchverfahren erreicht werden kann. Letztere Suchverfahren stellen Fallbeschreibungen und Suchanfragen als Vektoren von biomedizinischen Begriffen (Konzepten) dar, die aus Text- und/oder Bildmodalitäten mit Hilfe von Konzeptzuordnungsalgorithmen automatisch erstellt werden können. Wir schlagen ein Rahmenwerk für die multimodale Suche in medizinischen Fallbeschreibungen vor, das textbasierte Suchverfahren (inklusive Erweiterung von Suchanfragen) und konzeptbasierte Suchverfahren durch späte Fusion kombiniert. Wir zeigen, dass damit eine Steigerung der Sucheffektivität um 49% möglich ist, wenn praktisch einsetzbare Komponentensysteme durch lineare Fusion kombiniert werden. Das Potenzial einer weiteren Verbesserung wird experimentell als Effektivitätssteigerung von 166% gegenüber der Volltextsuche geschätzt, wobei eine adaptive Fusion von idealen Komponentensystemen betrachtet wird. Weitere wissenschaftliche Beiträge dieser Arbeit umfassen Vorschläge sowie die vergleichende Bewertung von Verfahren für die Konzeptzuordnung, für die Erweiterung von Suchanfragen und Dokumenten sowie für die automatische Klassifizierung und Aufteilung von zusammengesetzten Abbildungen, die in Fallbeschreibungen auftreten.

Abstract

Medical case retrieval (MCR) is defined as a multimedia retrieval problem, where the document collection consists of medical case descriptions that pertain to particular diseases, patients' histories, or other entities of biomedical knowledge. Case descriptions are multimedia documents containing textual and visual modalities (images). A query may consist of a textual description of patient's symptoms and related diagnostic images. This thesis proposes and evaluates methods that aim at improving MCR effectiveness over the baseline of fulltext retrieval. We hypothesize that this objective can be achieved by utilizing controlled vocabularies of biomedical concepts for query expansion and concept-based retrieval. The latter represents case descriptions and queries as vectors of biomedical concepts, which may be generated automatically from textual and/or visual modalities by concept mapping algorithms. We propose a multimodal retrieval framework for MCR by late fusion of text-based retrieval (including query expansion) and concept-based retrieval and show that retrieval effectiveness can be improved by 49% using linear fusion of practical component retrieval systems. The potential of further improvement is experimentally estimated as a 166% increase of effectiveness over fulltext retrieval using query-adaptive fusion of ideal component retrieval systems. Additional contributions of this thesis include the proposal and comparative evaluation of methods for concept mapping, query and document expansion, and automatic classification and separation of compound figures found in case descriptions.

Contents

CHAPTER 1

Introduction

The topic of this thesis is a specific problem in the field of multimedia retrieval, dealing with multimedia documents in the biomedical domain that represent medical cases. More detailed definitions, examples, and processes constituting medical case retrieval (MCR) are given in Section 1.1. The motivation for research on this topic (described in Section 1.2) emerges both from the application domain (clinical decision support systems) and from modest success of MCR systems known from literature. Details of the problem and research objectives investigated in this thesis are presented in Section 1.3. The relation between research objectives and the chapter structure of this document is explained in Section 1.4. Finally, Section 1.5 describes the scientific contributions of this thesis and concludes this introductory chapter.

1.1 Medical Case Retrieval

Medical case retrieval refers to the problem of finding case descriptions that are relevant to a given query in a large collection of medical cases. In general, a *case description* is a multimedia document describing a particular disease, patient's history, or other biomedical knowledge related to a certain medical case. The multimedia document usually contains a textual description (e.g. medical publication or diagnosis report) and a set of images or diagrams (e.g. diagnostic images), but may also include annotations with terms of a controlled biomedical vocabulary or audio recordings (e.g. speech or heartbeat sound). The *case query* is a multimedia document representing the user's information need. It may be as simple as a keyword, but it may also consist of a textual description of patient's symptoms and a corresponding set of diagnostic images and audio recordings.

Throughout this thesis, we focus on three modalities representing medical case descriptions and queries: (1) unconstrained *textual* descriptions in English language, (2) arbitrary digital images as *visual* data, and (3) annotations with *biomedical concepts* taken from a controlled vocabulary. Not all three modalities need to be available for a

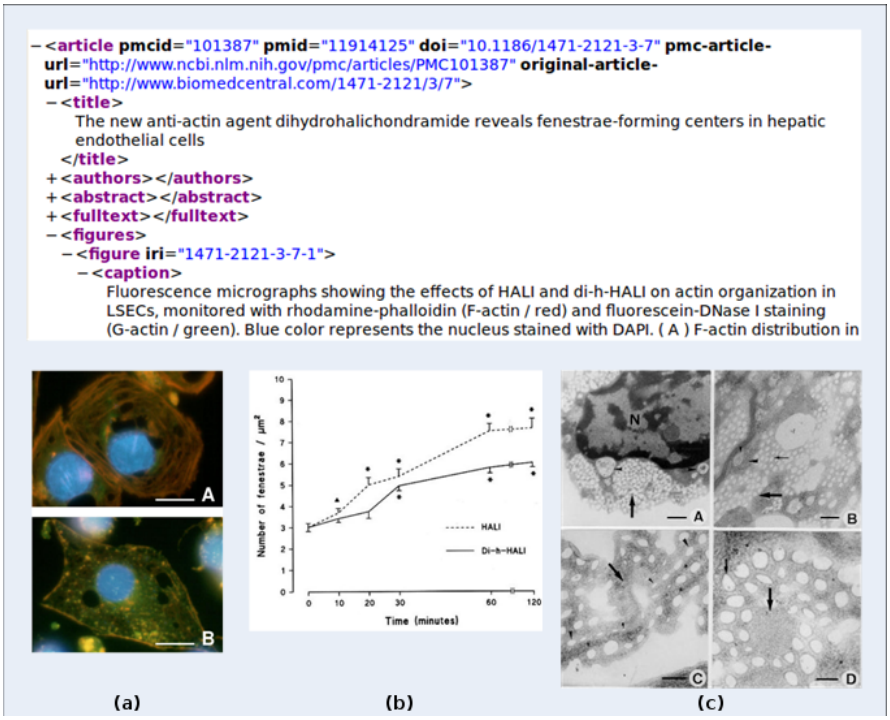


Figure 1.1: Example of a medical case description (scientific biomedical article) representing textual and visual modalities.

given case description or query; in particular, concept annotations are usually not given for queries, and are often incomplete or missing for case descriptions.

Figures 1.1 and 1.2 depict examples of a medical case description and query, respectively, represented by textual and visual modalities. The medical case description shown in Fig. 1.1 is actually a scientific biomedical article whose textual description is represented in XML format with separate fields for title, abstract, full text, and figure captions. Article figures represent the visual modality and are available as digital images (labeled (a), (b), and (c)) whose file names correspond to figure identifiers used in the XML description. Similarly, the medical case query shown in Fig. 1.2 consists of a textual description of patient's symptoms in XML format and three diagnostic images.

Note that some of the images appearing in Figures 1.1 and 1.2 consist of several subimages, but are stored as single images in digital image files. Such images are called *compound*. For example, article image (a) in Fig. 1.1 is a compound figure consisting of subfigures labeled A and B, respectively.

The main dataset used for experiments in this thesis is called *ImageCLEF MCR dataset* [104]. It consists of about 75,000 scientific articles from biomedical literature, including article images, and 35 queries representing patients' symptoms that comprise textual descriptions and diagnostic images. The dataset is not restricted to a particular

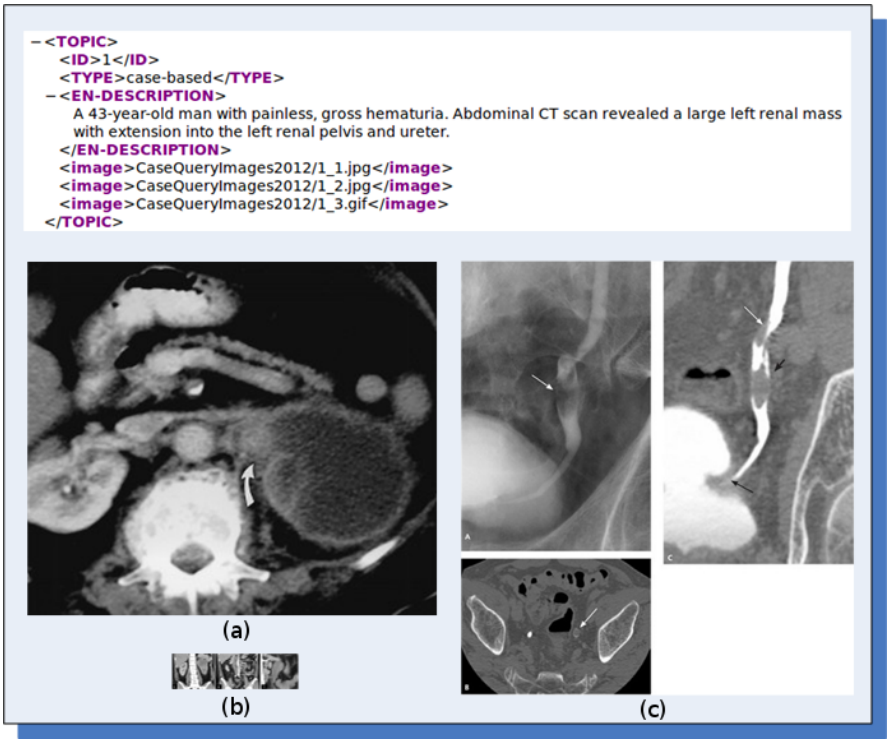


Figure 1.2: Example of patient's symptom description and diagnostic images that could be used as a medical case query.

medical domain and will be described in more detail in Chapter 3. The examples shown in Figures 1.1 and 1.2 are taken from this dataset.

Following the general information retrieval process [12], an MCR system first processes all case descriptions in a given collection (off-line phase) to produce an *index* that can later be used for efficient retrieval (see Fig. 1.3). The index is built from *features* extracted from textual, visual, or conceptual representations of case descriptions that allow to discriminate between cases within the collection. When a user then presents a case query to the system (on-line phase), the query is transformed to an internal representation (using a feature extraction method compatible with indexing) that can be compared efficiently to indexed case descriptions, allowing to produce a *ranked list* of case descriptions sorted by decreasing relevance with respect to the query.

Whether the top-ranked case descriptions returned by an automatic MCR system are indeed relevant to a given query needs to be judged by medical experts. The ImageCLEF MCR dataset comes with binary relevance judgments (*relevant* or *not relevant*) that were created by medical experts for a number of selected documents for each of the 35 queries. Relevance judgments can be used to evaluate a given MCR system by computing precision-recall-based metrics from the ranked list produced by the system. This evaluation method allows for comparing the effectiveness of different MCR systems or

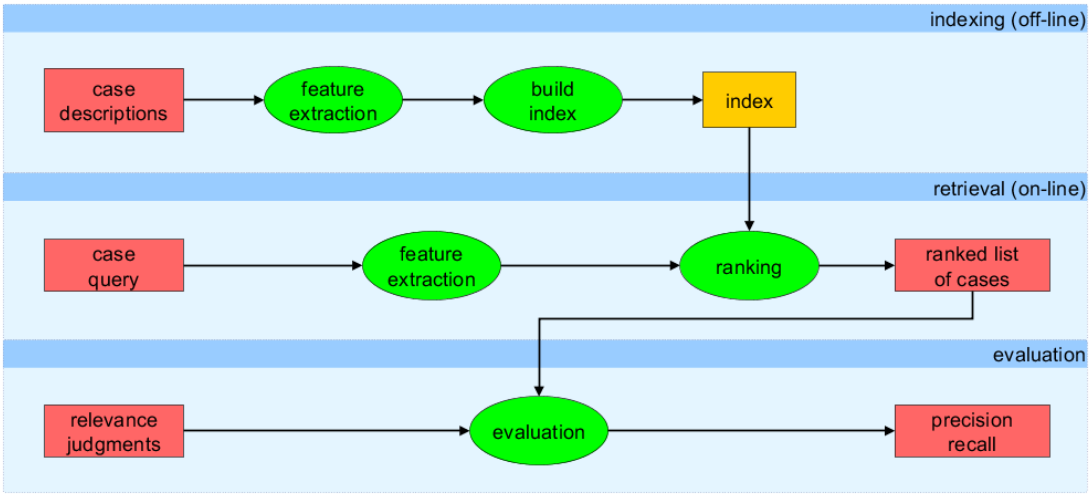


Figure 1.3: General processes of medical case retrieval.

measuring the change in effectiveness for certain modifications applied to a given MCR system. In the information retrieval research field, this system evaluation methodology has been used by the Text Retrieval Evaluation Campaign (TREC) [219] since the 1990s and dates back to the Cranfield paradigm [47] developed in the 1950s.

An MCR system is supposed to provide a similar functionality as classical information retrieval (IR) and content-based multimedia retrieval systems and, hence, may be built by using or combining existing technology for text and multimedia retrieval problems. However, constructing an effective MCR system faces some additional challenges that are not so pronounced or even do not occur with classical IR and general multimedia retrieval systems:

- A collection of medical case descriptions usually contains a large number of biomedical terms, where many of them are semantically related either by synonymy (semantic equivalence) or by hyponymy (*more-specific-than* relation). Classical IR methods may therefore miss or underestimate the relevance of documents containing only synonyms or hyponyms of biomedical terms given in the query.

- Biomedical terms may contain single letters or punctuation characters that are relevant, but are removed by indexing methods for classical IR. For example, the A in *Vitamin A* would be removed, because it is recognized as an English stop word.

- Two visually similar medical images may convey quite different semantic meaning due to important details varying between the two images. Such cases will degrade the effectiveness of most content-based image retrieval systems, because they use some form of visual similarity measure for relevance ranking.

- Conversely, semantically similar medical images may exhibit a large variance in visual appearance due to different image modalities (e.g. x-ray, ultrasound, CT, MR) or viewing perspectives (e.g. depending on the anatomical plane transecting a given organ).

Addressing these challenges provides a motivation for building an MCR system that is more effective than classical IR or general multimedia retrieval systems on collections of medical case descriptions. Other and more specific scientific motivations will be described in the next section.

1.2 Motivation

Clinical decision support systems provide clinicians with patient-specific assessments or recommendations to aid clinical decision making. Several features of such systems have been shown to improve clinical practice significantly [105]: automatic provision of decision support as part of clinician workflow, provision of recommendations rather than just assessments, provision of decision support at the time and location of decision making, and computer-based decision support. Depending on the degree of decision support expected from a computer system, these features may pose demanding requirements on the effectiveness and efficiency of used technology. Following the paradigm of *evidence-based medicine* [173], clinical decision making needs to integrate the physician's individual clinical expertise and the best available external clinical evidence from systematic research. Computer-based decision support systems may help to provide external evidence, learn from individual expertise, and possibly provide recommendations for diagnosis or treatment.

A well-known approach to designing a decision support system is the method of *case-based reasoning* (CBR), developed in the field of artificial intelligence research [1, 17]. Its main objective is to solve a new problem by applying previous experiences adapted to the current situation. For clinical decision support, the problem is represented by a patient's symptoms, and the solution is a decision about diagnosis and treatment. A problem and its solution are called a *case*, and cases are retained in a case library for subsequent reasoning about new problems. The process of case-based reasoning can be divided into four main tasks [1]: (1) for a given new problem, retrieve similar cases from the case library; (2) reuse the most relevant cases to propose a solution for the new problem; (3) revise the proposed solution to adapt it to the current problem; (4) retain the new case in the case library. Although successful CBR systems for different narrow medical application domains have been built and evaluated on a few hundred cases [17], general methods to design a CBR system applicable to larger and heterogeneous medical datasets still present an open research problem. This thesis addresses task (1) of a medical CBR system, as introduced in Section 1.1.

In addition to its use in decision support systems, medical case retrieval is also a relevant problem in medical education and research, because it allows to select interesting cases for students and to retrieve datasets for studies meeting case-based criteria.

Medical case retrieval tasks were issued on an almost yearly basis between 2009 and 2013 [88, 104] by the ImageCLEF evaluation campaign[1] [144], allowing researchers to evaluate their systems using a common large dataset. Over the years, the dataset evolved to a collection of about 75,000 unconstrained biomedical articles including contained figure images (see Chapter 3 for details). The retrieval performance achieved by the seven participating research teams in 2013 [88] reveals a similar pattern as in the 2012 task [145] (numbers in parentheses refer to the 2012 task, MAP = mean average precision [13]):

- best result achieved by textual retrieval only: MAP 24.3% (16.9%);
- best result using visual retrieval only: MAP 2.8% (3.7%);
- best result using combined textual and visual retrieval: MAP 16.1% (10.2%).

The observed pattern is established by the following facts: (1) content-based visual retrieval is by an order of magnitude less effective than textual retrieval; and (2) their combination is not able to reinforce the effectiveness of both techniques, contradicting the general expectation for fusing different retrieval methods. These observations confirm that medical case retrieval on general large datasets is still an open research problem. In particular, the obvious challenge of how to improve the effectiveness of an MCR system over purely textual retrieval is a key problem addressed by this thesis, as explained in the following section.

1.3 Problem Statement

Motivated by the fact that, according to the current state of the art, the most effective medical case retrieval systems employ purely textual retrieval techniques (see Section 1.2), the main research problem addressed by this thesis is how to improve MCR algorithms applied to general biomedical datasets containing textual and visual information. Starting from the observation that current content-based visual retrieval techniques perform poorly when used to retrieve medical cases, we hypothesize that utilizing a controlled vocabulary of biomedical concepts may help to bridge the semantic gap between relevance of medical cases and similarity of their textual and visual representations, leading to improved effectiveness of a suitable multimodal MCR approach.

[1] `http://imageclef.org/`

We consider two fundamental ways of introducing biomedical concepts into the retrieval process: (1) query or document expansion with biomedical concepts for text-based retrieval, and (2) representing medical case descriptions and queries by biomedical concepts for concept-based retrieval. For query and document expansion, biomedical concepts are added to the query or document text prior to applying text retrieval methods. Concept-based retrieval applies retrieval algorithms to vectors in concept space that represent medical case descriptions and queries. In addition, these two approaches may be combined by fusing the ranked list of documents returned by each of them (*late fusion*). We will apply linear fusion methods, which compute the rank/score of a document in the fused list as a linear combination of ranks/scores assigned by component systems, as well as query-adaptive fusion methods that choose combination weights based on predicted performance numbers of the component systems for a given query.

All these approaches give rise to the problem of finding biomedical concepts from a given controlled vocabulary that are relevant for a given medical case description or query, which can be considered as a multi-label classification problem with a large number of classes (several thousand biomedical concepts), called *concept mapping* in this thesis. Concept mapping algorithms need not only be evaluated with respect to classification accuracy, but also with respect to their effectiveness for the ultimate goal of medical case retrieval.

When using visual information (images) for concept mapping, further research problems may arise depending on the biomedical dataset used. The ImageCLEF MCR dataset (introduced in Section 1.1) consists of biomedical scientific articles treated as medical case descriptions, typically containing several article images with figure captions per document where roughly half of them are *compound figures* consisting of several subfigures. To obtain discriminative and semantically meaningful visual representations, concept mapping calls for an automated preprocessing of article images that recognizes and separates compound figures. Moreover, figure captions provide valuable textual information that can be utilized for concept mapping in addition to visual information.

The research objectives of this thesis are stated as follows, given in the order as treated in subsequent chapters:

O1 Design, implement, and evaluate an efficient and effective compound figure separation algorithm that is capable of processing at least 10 article images per second on current general-purpose hardware and achieves state-of-the-art separation accuracy.

O2 Select, apply, and evaluate existing techniques for mapping textual, visual, and multimodal medical case representations to biomedical concepts of a relevant controlled vocabulary. Evaluation is focused on measuring the ability of algorithms to reproduce ground-truth concept annotations.

O3 Improve text-based retrieval on the ImageCLEF MCR dataset by using results of concept mapping for query expansion, document expansion, and both.

O4 Evaluate concept-based retrieval on the ImageCLEF MCR dataset using results of concept mapping strategies studied for objective O2. Retrieval performance serves as another assessment criterion of concept mapping algorithms that is more relevant for MCR.

O5 Combine text- and concept-based retrieval by linear and query-adaptive fusion techniques.

The desired processing rate of the compound figure separation algorithm was chosen to allow for processing the approximately 300,000 images of the ImageCLEF MCR dataset in a reasonable amount of time (a few hours in a parallel processing environment). Concept mapping of images (represented by either visual or multimodal features) requires a postprocessing step to aggregate concepts of images belonging to the same medical case description or query, which needs to be incorporated into investigations for objective O2. Note that evaluation of concept mapping algorithms demanded by objective O2 may lead to different qualitative results than assessment by concept-based retrieval (O4), because ground-truth concept annotations may be incomplete or do not match the typical granularity of concepts produced by a certain concept mapping algorithm.

1.4 Structure of Document

Since the problem of medical case retrieval can be considered as a multimedia retrieval problem in the biomedical domain, we give an overview of the relevant literature on multimedia retrieval in Chapter 2. According to the technology utilized in our approaches to MCR, we summarize existing techniques for textual retrieval, content-based visual retrieval, and data fusion in information retrieval. Although traditionally not used in the context of multimedia retrieval, the research field of multi-view learning may support MCR by providing effective methods for mapping medical case descriptions to biomedical concepts. Hence, multi-view learning is reviewed in the same chapter in Section 2.6.

Chapter 3 describes the ImageCLEF MCR dataset used for experiments throughout the thesis, and presents our contribution to preprocessing article images corresponding to research objective O1 (see Section 1.3). The controlled vocabulary of biomedical concepts used for experiments is introduced in Chapter 4, which additionally elaborates on several methods for concept mapping, as expressed by research objective O2.

Following these preparatory chapters, three MCR approaches utilizing biomedical concepts are proposed and evaluated in separate chapters, corresponding to research

objectives O3, O4, and O5, respectively. The approach investigated in Chapter 5 aims at enhancing text-based retrieval techniques with biomedical concepts by document and query expansion. Using results of concept mapping methods, concept-based retrieval is evaluated in Chapter 6, thereby providing another evaluation metric for the effectiveness of concept mapping algorithms. The fusion of text-based and concept-based retrieval for improving MCR effectiveness is the objective of Chapter 7. Finally, Chapter 8 summarizes experimental results of the thesis and provides detailed suggestions for future work.

1.5 Contributions

The scientific contributions of this thesis emerge from pursuing the research objectives O1–O5 stated in Section 1.3. In particular, the following information is added to scientific knowledge in the fields of multimedia and biomedical information retrieval:

- Novel automatic methods for compound figure classification and separation [208, 209, 210] that are slightly more effective than existing automatic and semi-automatic approaches, while allowing a processing rate of 12 compound figures per second on current commodity hardware.
- A comparative evaluation of existing as well as new ad-hoc concept mapping strategies applied to associate medical case descriptions and queries with relevant biomedical (MeSH) concepts. Effectiveness of concept mapping is measured by both reproducing manual ground-truth annotations and performance of subsequent concept-based retrieval.
- A comparison of various query and document expansion methods [205, 207] utilizing MeSH concepts for text-based medical case retrieval. The best query expansion methods achieve state-of-the-art performance on the ImageCLEF MCR dataset without using large external text corpora.
- A novel framework for multimodal retrieval combining text- and concept-based retrieval that is able to improve over state-of-the-art retrieval performance on the ImageCLEF MCR dataset. Analysis of retrieval results reveals limitations of the ImageCLEF MCR dataset caused by pooling of relevance judgments.

Additionally, a concept mapping strategy based on multi-view learning has been designed, whose implementation and evaluation had to be postponed to future work due to time constraints. We nevertheless include the conceptual work on this topic in the thesis, because it seamlessly integrates with our study of concept mapping strategies and may serve as a basis for future scientific work.

The PhD proposal [204] has been presented and published at the Doctoral Symposium of ACM Multimedia 2014 conference [206], where it received the best Doctoral Symposium paper award.

CHAPTER 2

Multimedia Retrieval Background

This chapter provides an overview of literature and background information relevant for medical case retrieval (MCR). As MCR constitutes a rather young and narrow research field that applies and integrates many methods from other fields of artificial intelligence research, Section 2.1 explains these originating research fields and their dependencies. Since we consider MCR as a special multimedia retrieval problem, we base our presentation mainly on methods of *multimedia retrieval* research [121, 67], which again originated from other, more traditional fields of information retrieval (IR), most notably from the text retrieval, content-based visual retrieval, and information fusion fields.

We therefore present contributions of *text-based retrieval* (Section 2.2), *content-based visual retrieval* (Section 2.3), and *information fusion* (Section 2.4) to MCR as subfields of multimedia retrieval, although these research fields are usually not conceived as such. We deliberately ignore content-based audio retrieval, because this is not yet a subject of current research in MCR and not of this PhD project.

Since we hypothesize that the utilization of external biomedical knowledge may improve the effectiveness of MCR (see Section 1.3), another field of artificial intelligence research, namely *knowledge representation*, bears some relevance for MCR. Controlled vocabularies and ontologies of the biomedical domain, as results of knowledge representation research, and their use for MCR are reviewed in Section 2.5.

The chapter is concluded by a review on *multi-view learning*, which has not yet been applied to MCR, but provides methods that may be applied to the problem of mapping case descriptions to biomedical concepts (see Chapter 4). Moreover, some multi-view learning methods may be used to develop novel multimodal MCR techniques in future work (see Chapter 8). Portions of the text in this chapter have been reused from the PhD exposé [204].

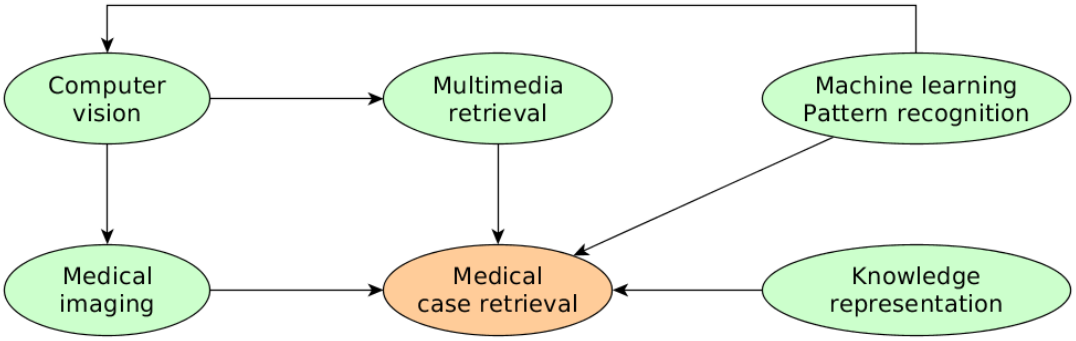


Figure 2.1: Research fields related to medical case retrieval.

2.1 Related Research Fields

The narrow research field of medical case retrieval (MCR) can be positioned at the intersection of five larger areas of artificial intelligence research, as depicted in Fig. 2.1:

- *Multimedia retrieval:* Indexing and retrieving multimedia documents requires techniques from classical information retrieval, hereafter called *text-based retrieval*, from content-based image and video retrieval, referred to as *content-based visual retrieval*, and from the *information fusion* literature dealing with the combination of several information retrieval systems or information sources.

- *Knowledge representation:* In an attempt to improve retrieval effectiveness on biomedical document collections, approaches incorporating *external knowledge* into the retrieval process have been proposed, often representing expert knowledge by biomedical ontologies or controlled vocabularies.

- *Computer vision:* When utilizing images for content-based retrieval, computer vision methods are needed to extract discriminative features and detect semantic concepts.

- *Medical imaging:* For diagnostic images, more specific techniques related to characteristic properties of medical images may be required to apply computer vision methods.

- *Machine learning and pattern recognition:* Many problems in computer vision, including detection of semantic concepts in images, require machine learning or pattern recognition techniques to achieve effective solutions. In the context of MCR, machine learning may help to find biomedical concepts relevant for a given case query or case description.

We note that multimedia retrieval, and hence MCR, usually involves user interaction when deployed in a real world setting. So the research fields of human-computer interaction and human-centered computing play an important role for designing a complete MCR system. However, the focus of this thesis is on automatic retrieval and system evaluation methods without user interaction, so these two research fields are ignored.

The following subsections give an overview of literature relevant for MCR in the research fields described above. The literature review is not complete, in particular we do not review the fields of computer vision and medical imaging explicitly, because their techniques are used in many publications related to (biomedical) content-based visual retrieval. For similar reasons, we do not review the vast literature on machine learning and pattern recognition [25, 84, 142], except for the subfield of multi-view learning (Section 2.6). However, we are confident that the presented overview reflects the current state of the art and does not miss substantial advances in the MCR field.

2.2 Text-Based Retrieval

Classical information retrieval (IR) [13] has been dealing with text retrieval for several decades, and a number of traditional techniques has proven to provide robust and efficient tools to perform text retrieval on general datasets. We focus on well-known "standard" models and techniques of text retrieval for two reasons: (1) to the best of our knowledge, there is no recent technique for text-based retrieval on general medical datasets that performs substantially better than traditional text retrieval methods; and (2) evaluation and comparison with other approaches becomes more meaningful if they are based on well-known IR models. Moreover, text-based IR methods play an important role for retrieval of health care and biomedical information [91].

Section 2.2.1 gives an overview of well-known IR models with an emphasis on the model used for experiments in this thesis. Measuring retrieval performance of IR systems is the subject of Section 2.2.2. Note that the same evaluation methodology is often used to evaluate multimedia retrieval systems. Section 2.2.3 presents a more detailed survey on query expansion methods, as they are used to utilize biomedical concepts for text-based retrieval in Chapter 5.

2.2.1 Information Retrieval Models

As described in many textbooks on information retrieval (e.g. [13, 135, 162]), two standard models of text retrieval are the *vector space model* [176] and the *probabilistic model* [166], combined with TF-IDF (term frequency, inverse document frequency) [163,

198, 228] or BM25 [164] term weighting. These methods are able to deliver state-of-the-art text retrieval performance, and mature open-source implementations are available, most notably Lucene[1] and Indri[2] [138].

There are several alternative information retrieval models that can be classified into set-theoretic, algebraic, and probabilistic models [13]. Two prominent alternative probabilistic models are language models [126, 168] and divergence from randomness [5]. The latter has been found to be the most effective model on a biomedical dataset [2]. However, due to the lack of available implementations we do not consider these models for experimental evaluation.

Experiments in this thesis use Lucene version 4.10.2 with its default implementation of the vector space model[3]. Lucene defines a variant of TF-IDF weighting $w(t,d)$ of term t in document d as:

$$w(t,d) = \sqrt{\mathrm{TF}(t,d)} \cdot \left(1 + \log \frac{N}{\mathrm{DF}(t)+1}\right) \cdot \mathrm{Norm}(d) \tag{2.1}$$

where $\mathrm{TF}(t,d)$ denotes the number of occurrences of term t in document d (*term frequency*), N is the number of documents in the dataset, $\mathrm{DF}(t)$ is the number of documents in the dataset that contain term t (*document frequency*), and $\mathrm{Norm}(d)$ is a normalization factor involving document length. The second factor is known as *inverse document frequency*. The document norm $\mathrm{Norm}(d)$ depends on document length $\mathrm{len}(d)$ (number of indexed words in d) as $1/\sqrt{\mathrm{len}(d)}$. When matching a given query q to an indexed document d, Lucene multiplies query and document weights to compute a relevance score:

$$s(q,d) = \mathrm{Coord}(q,d) \cdot \mathrm{Norm}(q) \cdot \sum_{t \in q} \mathrm{Boost}(t) \cdot w(t,d) \tag{2.2}$$

where $\mathrm{Coord}(q,d)$ is a factor involving the ratio of query terms contained in document d, $\mathrm{Norm}(q)$ is a normalization factor for query length (as defined earlier), $\mathrm{Boost}(t)$ is a user-supplied search-time boosting factor of query term t, and $w(t,d)$ is the weight of term t in document d defined in (2.1). The boosting factor $\mathrm{Boost}(t)$ of query terms can be specified by the user formulating the query. Lucene's query syntax[4] allows to denote the boosting factor f for a query term t as $t\hat{}f$ (e.g. term^1.2).

[1] `http://lucene.apache.org/`

[2] `http://www.lemurproject.org/indri/`

[3] See Java class `org.apache.lucene.search.similarities.TFIDFSimilarity` in API documentation at `http://lucene.apache.org/core/4_10_2/core/`

[4] `https://lucene.apache.org/core/4_10_2/queryparser/org/apache/lucene/queryparser/classic/package-summary.html`

2.2.2 Retrieval Evaluation

Text retrieval evaluation is an established methodology developed since the 1950s, starting with the so-called Cranfield experiments and further developed in the context of Text Retrieval Conferences (TREC) until today [13, Chap. 4][219]. The resulting system evaluation method, also known as *TREC-style evaluation*, aims at assessing the quality of ranked lists of documents retrieved by a system under test for a given set of queries, resulting in measurement of the system's *retrieval performance*. As retrieval performance is determined from ranked lists, obtained numbers are related to retrieval effectiveness only, not to run-time efficiency of the retrieval process. Note that such an evaluation does not pay attention to user satisfaction or user interactions during a search session, except for measuring a system's ability to rank relevant documents near the top of the retrieved list. TREC-style evaluation is therefore suited for automatic retrieval systems and provides objective criteria allowing to compare the retrieval performance of different systems.

TREC-style evaluation requires the preparation of a *dataset* consisting of a large corpus of text documents, a set of queries, and *relevance judgments* stating which documents of the corpus are relevant or irrelevant for a given query (also called *ground-truth judgments*). Since relevance judgments are usually created by human experts, it is infeasible to judge all documents of a large corpus for relevance to a given query. Documents are therefore selected for judgment by *pooling* documents retrieved by several retrieval systems. For example, a pooling strategy could select 100 (top-ranked) retrieved documents from each of three retrieval systems for a given query. Pooled documents are then presented to human experts for relevance judgment.

A retrieval system under test is required to index all documents of the dataset and then process each prepared query to produce a ranked list of retrieved documents. Relevance judgments must not be made available to the tested system, but are used afterwards to compute retrieval performance measures from retrieved lists of documents.

From the many different retrieval performance measures proposed in literature [13, Chap. 4], we focus on the commonly used class of measures based on *precision* and *recall*, which will be used for retrieval experiments throughout this thesis. More specifically, retrieval performance will be measured by *mean precision at n* (for an integer $n > 0$, usually $n = 5$ or $n = 10$), denoted as P@n, and by *mean average precision* (MAP). To define these measures, let Q be the query set defined by the prepared dataset, let R_q be the list of documents retrieved by the tested system for query $q \in Q$, let G_q be the set of relevant documents for query q contained in ground-truth judgments, and let $\mathrm{Rel}(d)$ be a binary function representing the relevance judgment for document d, where $\mathrm{Rel}(d) = 1$ if d has been judged as relevant and $\mathrm{Rel}(d) = 0$ otherwise (i.e. d has not been judged or judged as irrelevant). P@n, MAP, precision π_q, and recall ρ_q (for query $q \in Q$) are then defined by the following sequence of equations:

$$P_q = \{n \mid 1 \leq n \leq |R_q|,\ \mathrm{Rel}(R_q[n]) = 1\} \tag{2.3}$$

$$\pi_q = \frac{|P_q|}{|R_q|} \tag{2.4}$$

$$\rho_q = \frac{|P_q|}{|G_q|} \tag{2.5}$$

$$P@n(q) = \frac{1}{n}\sum_{i=1}^{n} \mathrm{Rel}(R_q[n]) \tag{2.6}$$

$$P@n = \frac{1}{|Q|}\sum_{q \in Q} P@n(q) \tag{2.7}$$

$$AP_q = \frac{1}{|P_q|}\sum_{n \in P_q} P@n(q) \tag{2.8}$$

$$MAP = \frac{1}{|Q|}\sum_{q \in Q} AP_q \tag{2.9}$$

$R_q[n]$ is the document retrieved at rank position n, and P_q is the set of rank positions of retrieved relevant documents, hence $|P_q|$ is the number of retrieved relevant documents. Precision π_q is the fraction of relevant documents within all retrieved ones, whereas recall ρ_q is the fraction of retrieved documents within all judged relevant ones. $P@n(q)$ is the precision at n for query q, and AP_q is the average precision for q, which can be regarded as a measure combining precision and recall. Note that P@n and MAP are just mean values of $P@n(q)$ and AP_q, respectively, over all queries $q \in Q$. Moreover, mean values of π_q and ρ_q over the query set are referred to as precision π and recall ρ.

Precision, recall, P@n, and MAP are constrained to values in the range $[0, 1]$, and an ideal retrieval system would achieve the value 1 for all performance measures. However, practical IR systems often display a tradeoff between precision and recall: if the system is tuned for high precision, recall will decrease, and vice versa. Retrieval performance is therefore usually reported using multiple measures, but since MAP combines precision and recall, it is often used as the main performance measure for comparing the effectiveness of IR systems.

2.2.3 Query Expansion

A fundamental limitation of retrieval performance of textual information systems is the mismatch of words used to express the same concepts in the query and in the document collection, known as the *vocabulary problem* in information retrieval. One methodology to address this problem, called *query expansion* (QE), is to automatically expand the user's query with words related to the user's information need (i.e. the query *topic*) before sending the query to the retrieval system. From the variety of QE

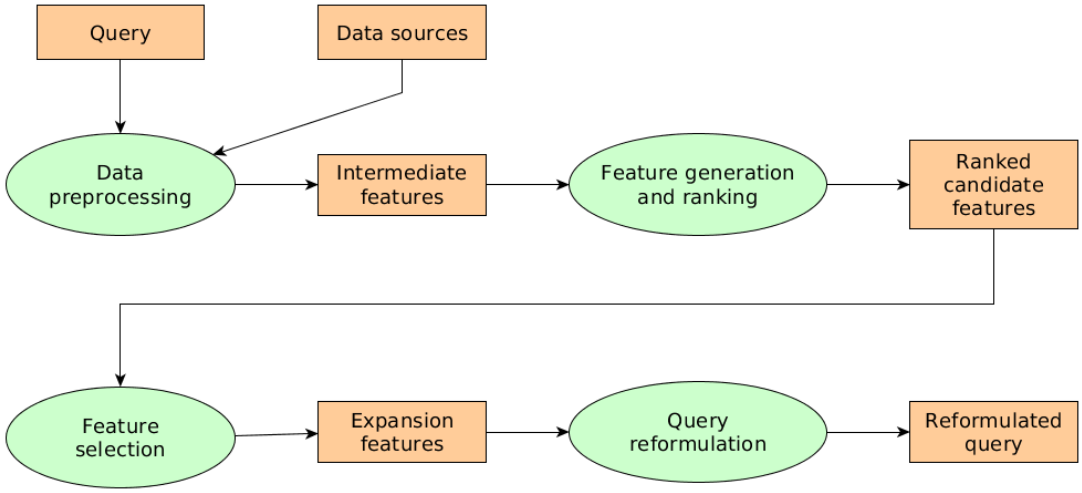


Figure 2.2: Stages of query expansion process [34].

techniques proposed during the last four decades, we try to summarize the key methods and principles, as described and classified by Carpineto and Romano [34].

Alternative methodologies to overcome the vocabulary problem are interactive query refinement (e.g. [13]), relevance feedback [172], word sense disambiguation [147], and search results clustering [33]. The first two alternatives cannot be applied to the MCR task covered by this thesis, as they require interactive user input. Word sense disambiguation techniques do not seem to provide any advantages over QE with respect to effectiveness and efficiency of information retrieval [3, 34], so they have not been investigated in this work. Search results clustering has typically been employed for browsing through web search results and does not seem to be beneficial for the automatic MCR task and comparably small dataset considered here.

Query expansion works by leveraging external or in-collection data sources to generate and select expansion features used to reformulate the original query. A general process pipeline common to all QE techniques proposed so far consists of four stages (Fig. 2.2): (1) preprocessing of data sources, often performed at indexing time; (2) generation and ranking of candidate expansion features; (3) selection of expansion features; and (4) query reformulation.

2.2.3.1 Query Expansion Process

To illustrate the process pipeline and to describe a QE method used in our experiments, consider the following simple pseudo-relevance feedback approach inspired by Rocchio's relevance feedback method [34, 167]. An inverted index implementing a vector space model using TF-IDF weights is used initially to retrieve a ranked list of documents matching the original query. This list of documents acts as data source for QE, and

stage (1) of the process pipeline needs to ensure that the inverted collection index allows to access the TF-IDF weights of terms. The TF-IDF weights of every term (word) in the n top-ranked documents are summed up, and terms are sorted by their accumulated weight. Initial retrieval and sorting terms of top retrieved documents represent stage (2). Finally, the first k terms of the sorted list (stage (3)) are added to the original query (stage (4)).

From the four process pipeline stages, feature generation and ranking (2) is the most critical one and gave rise to a large variety of proposals in the literature. We try to identify the key approaches in Section 2.2.3.2. The feature generation method determines the required preprocessing (1), and the ranking method enables or disables certain feature selection techniques (3). The following two paragraphs give an overview of existing methods for feature selection (3) and query reformulation (4).

Feature selection Selecting the first k features is always possible, and there is empirical evidence that a value of k between 10 and 30 is a good choice for many general datasets, because retrieval performance decreases only slowly for sub-optimal values of k [34]. When the feature scores allow for consistent semantic interpretation (e.g. as probabilities), features with a score greater than a certain threshold can be selected. It is known that, on average over many queries, a rather large fraction of terms selected by these simple approaches are harmful to retrieval performance [31]. Several advanced feature selection methods have been proposed to improve the fraction of relevant expansion terms for a given query, including the combination of multiple term ranking functions [35], generating multiple feedback models by resampling documents and varying the query [51], choosing k as a function of the ambiguity of the (Web) query [45], employing supervised learning to discriminate between relevant and irrelevant expansion terms [31], and solving an optimization problem with respect to uncertainty sets [49].

Query reformulation The simplest method for query reformulation (4) is to add the selected expansion features to the original query without modifying their weights. The most common approach, however, is to give different weights to terms of the original query and to expansion terms, and to incorporate the score of expansion features computed in stage (2). A general formulation based on Rocchio's reweighting formula for relevance feedback [167, 175] is the following.

$$w'_{t,q'} = (1 - \lambda) \cdot w_{t,q} + \lambda \cdot s_t \cdot w_{t,Q} \tag{2.10}$$

Here $w_{t,q}$ and $w_{t,Q}$ are the weights assigned by the underlying retrieval system to term t within the original query q and within the sequence Q of expansion terms, respectively. s_t is the term score computed in stage (2), λ is a parameter ($0 \leq \lambda \leq 1$) to set the

relative importance of expansion terms with respect to original query terms, and $w'_{t,q'}$ is the modified weight of term t in the expanded query q'. If the order of magnitude of expansion term scores s differs from 1, normalization is needed [227]. Alternatively, the values s_t can be computed from an inverse function of term ranks produced in stage (2) [35, 93].

Although giving expansion terms a fixed lower importance than original query terms (e.g. $\lambda = 0.3$) is common practice, a query-specific value of λ can also be predicted by supervised learning in a pseudo-relevance feedback setting [132]. Alternatively, a parameter-free query reweighting method has been proposed [4].

When expansion features are generated using a thesaurus or ontology, score values s_t may accommodate properties and relationships of nodes in the term network [102], or the importance factor λ may depend on the type of such properties and relationships [218].

In language modeling approaches of information retrieval [13, 126], query reweighting arises naturally by smoothing the probability distribution of query terms (*query model* θ_q) with that of query expansion terms (*query expansion model* θ_Q), in analogy to smoothing the document model with the collection model [244]. When applying the Jelinek-Mercer interpolation [100] to smoothing the query model, the probability distribution of the final expanded query model is given by

$$p(t|\theta'_q) = (1 - \lambda) \cdot p(t|\theta_q) + \lambda \cdot p(t|\theta_Q), \tag{2.11}$$

which is analogous to reweighting formula (2.10).

A more general approach to query reformulation is to use Boolean [78] or structured queries [50], or the advanced query formulation features of recent query languages like Indri[5], as proposed in [9] for instance.

2.2.3.2 Query Expansion Approaches

Following and extending the classification of Carpineto and Romano [34], we give an overview of known query expansion techniques according to the conceptual paradigms used to generate expansion features (stage (2) of the query expansion process, Figure 2.2). For each class of techniques, we try to identify the key approaches characterizing the main ideas and results of its class.

We can distinguish five classes of query expansion approaches: (i) those based on linguistic analysis, (ii) corpus-specific global techniques, (iii) query-specific local techniques, (iv) approaches using external knowledge models, and (v) other innovative techniques that do not fit into the former classes. The following paragraphs review the approaches of each class in more detail.

[5] `http://www.lemurproject.org/indri/`

Linguistic Analysis Approaches applying linguistic analysis use morphological, lexical, syntactic, or semantic word relationships to generate expansion features from query words. A frequently used technique is stemming [96, 110, 154], which replaces inflected or derivational forms of a word by its stem, usually at indexing time. Syntactic analysis has been used to derive relationships between parse trees of query and top-ranked passages, in order to learn the most relevant relations for the query [200]. Semantic associations of words are often represented by thesauri or ontologies, which are the subject of class (iv).

Corpus-specific global techniques These techniques use information extracted from the the entire collection of documents during the pre-processing stage to derive associations between the query and candidate expansion features. Early approaches exploited term co-occurrence at the document or passage level, but could not consistently improve retrieval performance [141]. Two successful key strategies are term concepts [155] and term clustering [15, 54, 181]. *Term concepts* are vector representations of terms indexed by document weights, which can be viewed as a dual representation of the standard document vector space model. The query is represented as a linear combination of term concepts and compared to indexed term concepts by cosine similarity. The resulting ranked list of expansion term candidates is supposed to be more relevant to the whole query than to individual query terms.

The *term clustering* approach of Crouch and Yang [54] clusters documents by cosine similarity and assigns low-frequency terms of clusters to term classes, which are used as synonym classes for query expansion. Schütze and Pedersen [181] efficiently construct a thesaurus of terms sharing neighbors in the document corpus (second-order co-occurrence) by iterative clustering of columns of co-occurrence submatrices, followed by an SVD decomposition that allows to represent terms by dense 20-dimensional real-valued vectors. However, the authors do not use the thesaurus directly for query expansion (although this would be possible), but perform retrieval on document representations derived from term vectors (context vectors).

The advantage of global techniques, namely the generation of potentially discriminative features for query expansion, is also their main limitation: features that co-occur frequently in the document collection may be irrelevant for the given query.

Query-specific local techniques The aforementioned problem is addressed by query-specific local techniques, which aim at utilizing the local context provided by the query for expansion. Usually top-ranked documents retrieved in response to the original query (also called *pseudo-relevant documents*) are analyzed to generate expansion features. A simple and well-known method, inspired by Rocchio's relevance feedback technique [167], is *pseudo-relevance feedback*, where collection-based term weights (e.g. TF-IDF

weights) are collected from pseudo-relevant documents and used to rank terms as expansion candidates. However, the effectiveness of this approach may be limited by the fact that top-ranking terms may not be relevant for the query, although discriminative for the entire collection.

More advanced local key approaches are analysis of *feature distribution difference*, *query language modeling* and *document summarization*. The former derive term-ranking functions from measuring the term distribution difference between the set of pseudo-relevant documents and the entire collection. Well-known instances of term distribution difference models are the binary independence model [166], the chi-square distance [65], Robertson's selection value [165], and the Kullback-Leibler distance [32]. More term-ranking functions and an experimental study comparing different methods are reported by Wong et al. [227].

Query language modeling approaches estimate a term probability distribution (language model) for the query and consider the most likely terms for query expansion. The query language model is typically estimated using pseudo-relevant documents, as is done by the two main representatives: the *mixture model* [243] and the *relevance model* [116]. The former considers the likelihood of pseudo-relevant documents as a mixture of the query topic model and the collection language model. The query topic model is estimated using the expectation-maximization algorithm [58] as to maximize the likelihood of pseudo-relevant documents. The relevance model assumes that both the query and pseudo-relevant documents are samples from the same unknown term probability distribution $p(t|\theta_R)$ (θ_R is the relevance model). Using the conditional probability of term t given that the original query words have just been observed, an efficient expression for estimating $p(t|\theta_R)$ from pseudo-relevant documents can be derived. Metzler and Croft [139] propose an important generalization of the relevance model that incorporates term dependencies and proximity-based features by modeling the joint distribution of query and relevant document by Markov random fields.

Document summarization techniques preprocess pseudo-relevant documents to represent them by more compact and informative features before applying a term-ranking function. *Local context analysis* [234] uses term-concept co-occurrence extracted from passages (text windows of fixed size) of pseudo-relevant documents, where a concept is a group of adjacent nouns. Other approaches use text summarization techniques [115] or intra-document feature clustering and classification [38].

External knowledge models Query expansion techniques using external knowledge models utilize linguistic or domain-specific information not already available in the document collection, but in external knowledge representations like thesauri or ontologies (see [180] for a discussion on the distinction between these concepts). Ontology-based query expansion is analyzed in [148] and reviewed in [22].

A well-known linguistic thesaurus is WordNet[6] [140], which has frequently been used to find synonyms and related words of query words for general collections [77, 134, 218]. The major problem with the use of WordNet is word sense disambiguation [147], which has been addressed by several advanced approaches [76, 124, 195].

The semantic relationships between concepts defined in knowledge models may be used to generate query expansion features based on their conceptual distance in the semantic network. Liu et al. [127] rank key phrases extracted from pseudo-relevant documents according to their conceptual distance to the query phrase on WordNet. Tudhope et al. [214] assign traversal costs to the relationships in a domain-specific thesaurus and generate expansion concepts by traversing the semantic network until a predefined cutoff distance threshold is reached. Candidate concepts are ranked by their average conceptual distance to all query terms.

In the medical domain, many ontologies and thesauri have been developed to store and classify medical knowledge [16, 61] (see Section 2.5). Query expansion using the MeSH (Medical Subject Headings) thesaurus has been applied to medical case retrieval with varying success. Diaz-Galiano et al. [61] observed a significant increase in retrieval performance on the ImageCLEF 2005 and 2006 MCR datasets, whereas Mata et al. [136] could not using the ImageCLEF 2011 dataset. However, the latter authors reported a more successful approach in [53].

Other techniques There are some other principled approaches that do not fit into the classes described above. Collins-Thompson and Callan [50] construct a query-specific term network whose relations can be generated from various sources (WordNet, stemmer, external corpus, top retrieved documents) and are assigned transition probabilities. The term network is modeled as a Markov network, and terms with highest probability according to the stationary distribution are selected for expansion. Riezler et al. [161] apply supervised machine learning to translate the query to semantically related phrases, and extract expansion terms from them.

2.3 Content-Based Visual Retrieval

Datta et al. [57] give a comprehensive overview of research on *content-based image retrieval* (CBIR) of the previous decade. The authors define CBIR as "any technology that in principle helps to organize digital picture archives by their visual content". The search paradigm most commonly considered in CBIR research contributions is *query by example*, meaning that an image is available to be used as a query to retrieve relevant "similar" images from a large picture archive. Typically, the user of a CBIR system expects a semantic similarity of images relevant to the query, which depends on the

[6] `http://wordnet.princeton.edu/`

user context and application domain and may not be directly related to the visual similarity of images. This discrepancy is known as the *semantic gap* [191], which is still an open problem in many application domains of CBIR.

The medical imaging domain provides some opportunities that may help bridging the semantic gap, like better defined imaging semantics, rich metadata, and existing knowledge representations. But there are also additional challenges like its interdisciplinary nature, integration of different information sources, and limited availability of training data [251]. A review of CBIR in medical applications and its clinical benefits is given by Müller et al. [146].

From the many facets of CBIR research identified by Datta et al. [57], we focus on the core techniques supporting the basic CBIR process: (1) *feature extraction* represents an image by one ore more vectors of numbers capturing visual properties that are able to discriminate between relevant and non-relevant images, but are also invariant under irrelevant image transformations (e.g. rotation); (2) pattern recognition techniques are used to build *visual signatures* from feature vectors that reduce their dimensionality and aim at representing the desired image semantics, in an effort to bridge the semantic gap; (3) *similarity measures* are applied to visual signatures in order to retrieve (and rank) images that are most similar to a given query image.

A wealth of different image features and corresponding extraction algorithms has been proposed for CBIR [57]. Deselaers et al. [60] performed extensive experimental comparisons between 19 image features on different datasets, including the IRMA (Image Retrieval in Medical Applications[7]) 2005 dataset of 10,000 medical images. Feature types can be categorized into *global features* describing the visual properties of the entire image by a single feature vector, and *local features* extracted from certain locations or regions in the image. The visual properties captured by feature extraction methods include color, texture, and shape, and many proposed image features represent a combination of these. Among other mathematical models, wavelet transforms are used to represent texture features [64]. A more recent composite image descriptor capturing brightness and texture characteristics for medical image retrieval has been proposed by Chatzichristofis et al. [39].

Whereas global feature vectors are often used directly as visual signatures, local feature vectors of an image need to be summarized to form a signature. The *bag of features* approach applies clustering of local feature vectors of an image collection to construct a codebook of cluster centers (*visual words*), and every image is represented by a term vector of visual words [190], in analogy to text retrieval. Iakovidis et al. [97] build the visual signature by clustering wavelet coefficients and estimating the distributions of clusters using Gaussian mixture models and an expectation-maximization algorithm. They obtain promising medical image retrieval results on the IRMA dataset. Quellec et

[7] `http://irma-project.org/index_en.php`

al. [157] extend the wavelet-based visual signatures of Do and Vetterli [64] by adapting the wavelet basis in order to optimize retrieval performance for a given image collection. They evaluate their approach successfully on two specific homogeneous medical image datasets as well as on a face image dataset.

Another attempt to reduce the semantic gap is to express visual signatures in terms of *semantic concepts* automatically detected in images using pattern recognition techniques. A comprehensive and detailed discussion of concept-based video retrieval is given by Snoek and Worring [193]. Most of the techniques described there can also be applied to image retrieval. A well-known categorization scheme for diagnostic images is the *IRMA code* [120], classifying the visual content along four dimensions: image modality (e.g. X-ray, ultrasound, CT, MR), body orientation, body region, and biological system. IRMA categories may serve as concepts to build semantically meaningful visual signatures.

Rahman et al. [159] proposed a concept-based image retrieval framework utilizing class probabilities of multiple classifiers as visual signatures and cosine similarity for retrieval. Class probabilities are estimated from binary SVM classifiers. For different low-level visual feature spaces, concept-based similarity values are calculated separately and fused using a linear combination scheme where weights are optimized adaptively for each query. Weight optimization incorporates automatic relevance estimation based on classifier fusion over low-level feature spaces, but may also include user relevance feedback. The framework was evaluated on the ImageCLEF 2006 medical dataset using 116 IRMA categories and 4 low-level visual features (MPEG-7 Edge Histogram and Color Layout, GLCM-based texture features, and block-based gray values). In 2011, the authors proposed a similar retrieval scheme [158].

The visual signature of a query image needs to be compared to that of images in the collection to retrieve the "most similar" ones. The underlying assumption is that similarity of visual signatures is correlated with semantic relevance. Failure of this assumption indicates that the semantic gap has not been bridged successfully. Similarity of visual signatures is computed by applying an appropriate *similarity measure*. Eidenberger [66] conducted an extensive experimental comparison and analysis of many well-known similarity measures used for CBIR.

Güld et al. [80] describe a generic *framework for medical image retrieval systems* developed by the IRMA project [119]. The framework aims at enabling flexible and efficient development and deployment of retrieval algorithms in a distributed environment with web-based user interfaces. Demo applications using this framework are available online[8].

Zhou et al. [251] propose a framework for content-based medical image retrieval on a semantic level. They emphasize the need for a scalable semantic retrieval system

[8] `http://irma-project.org/onlinedemos_en.php`

(e.g. easily adaptable to different image modalities and anatomical regions) and for incorporating external knowledge. An architecture for integrated (symbolic and sub-symbolic) image feature extraction and semantic reasoning is proposed. As a prototype implementation, they describe a semantic anatomy tagging engine called ALPHA, which employs a novel approach to deformable image segmentation by combining hierarchical shape decomposition and CBIR.

A Java library supporting content-based text and image retrieval is LIRE[9] [130, 131]. It provides a number of different global and local image feature extractors and efficient indexing techniques for images and text based on Lucene[10].

During the last decade, *deep learning* techniques [117] were applied to large-scale computer vision problems with great success [109], causing a revolution in computer vision that made convolutional neural networks (CNNs) the dominant approach for visual recognition and detection tasks. Recently, recurrent neural networks were applied to automatically generate caption text of images [217]. Although such methods seem suitable for automatically mapping case descriptions to biomedical concepts, we did not consider deep learning methods in this thesis, because their successful application requires large amounts of training data in the biomedical domain that were not available to us.

2.4 Information Fusion

Information fusion (also known as data fusion or meta-search) is a well-known research field of information retrieval. The main objective is to combine multiple information sources to improve retrieval performance. Depending on the phase of the retrieval process chain where the combination happens, different *fusion levels* can be distinguished [215]: signal level, feature level, and decision level. Signal- and feature-level fusion are also called *early fusion*, whereas decision-level fusion is also known as *late fusion*, which aims at combining the results of multiple retrieval systems.

In the context of MCR, late fusion is of particular interest, because it allows for *multimodal fusion* of text and visual retrieval systems. Late fusion approaches can be categorized into *score-based* and *rank-based* methods, according to which information from retrieval result lists is used (score or rank). Wu [229] gives a concise overview of known methods of both categories and proposes a new weight optimization method for linear score combination based on the multiple linear regression technique. Moreover, the author addresses another important issue of score-based data fusion systems, namely how to obtain reliable scores from score or rank information provided by component systems (*score normalization*). The logistic and cubic regressions models are found to

[9] `http://www.semanticmetadata.net/lire/`
[10] `http://lucene.apache.org/`

provide reliable solutions to the score normalization problem. The proposed approach is evaluated on several text retrieval datasets of recent TREC (Text Retrieval Conference [219]) challenges.

Zhou et al. [249] investigated and generalized the classical score combination methods combMAX, combSUM, and combMNZ [72] for single-modal and multimodal fusion of the 8 best runs submitted to the ImageCLEF medical image retrieval tasks in 2008 and 2009. They conclude that logarithmic rank penalization is the most stable score normalization strategy, but there is no significant difference between the various score combination methods considered.

Gkoufas et al. [73] evaluated linear combination methods using multi-field textual retrieval and visual retrieval built into LIRE [130] on the MCR datasets of ImageCLEF 2009 and 2010. Fusion of textual and visual retrieval could not improve retrieval performance (MAP) over fulltext-only retrieval on the ImageCLEFmed 2009 and 2010 datasets, only early precision (at 5 and 10) increased slightly.

A different approach to information fusion is *filtering*, where component retrieval systems are used in a pipeline fashion such that a subsequent system works on a reduced dataset that has been filtered by the previous system (i.e. documents supposed to be irrelevant have been filtered out). Usually, filtering is applied in combination with other fusion techniques. Such an approach for medical image retrieval using an IRMA code classifier for filtering was proposed by Rahman et al. [158]. A more general approach using text retrieval as the filtering stage and locality-sensitive hashing for visual retrieval was proposed by Zhang et al. [247].

2.5 Knowledge Representation

In the context of information retrieval, *external knowledge* is an information source that is not available in the dataset or query, but could be utilized to improve retrieval performance. There are two main techniques to achieve this aim: *query expansion* (see Section 2.2.3) and *multi-label annotation* [213, 246] of documents. Both techniques may incorporate external knowledge in the form of *controlled vocabularies* or *ontologies*[11], which specify concepts, relationships, and other distinctions that are relevant for modeling a domain.

In the biomedical domain, some well-known controlled vocabularies include Medical Subject Headings[12] (MeSH) used to index literature [123, 128], the Gene Ontology[13] (GO) modeling biological systems, and RadLex[14] providing radiology terms used to

[11] See e.g. [171, 180] for a distinction between the notions of controlled vocabularies, formal ontologies, and other ontological artifacts.

[12] `https://www.nlm.nih.gov/mesh/`

[13] `http://www.geneontology.org/`

[14] `https://www.rsna.org/RadLex.aspx`

annotate medical images. More comprehensive ontologies representing encyclopedic knowledge in medicine are the Foundational Model of Anatomy[15] (FMA) and the Systematized Nomenclature of Medicine, Clinical Terms[16] (SNOMED-CT). Due to the growing number of ontological resources in the biomedical domain, several efforts to aggregate and link multiple ontologies and vocabularies have been made, resulting in meta-vocabularies or combined databases, including the Unified Medical Language System[17] (UMLS) meta-thesaurus, the Entrez database provided by the National Center for Biotechnology Information[18] (NCBI) [150], and the BioPortal[19] database of the National Center for Biomedical Ontology (NCBO).

Query expansion using external knowledge models has already been covered by Section 2.2.3.2. Multi-label annotation employs machine learning techniques to automatically assign several, possibly related semantic concepts to (multimedia) documents. This can improve retrieval performance if the annotated concepts are relevant to the query and add information to documents (i.e. the annotated label is not already contained in text documents). If the possible labels are organized in a tree structure, multi-label classification specializes to *hierarchical multi-label classification* (HMC). Dimitrovski et al. [63] propose an HMC classifier for medical image annotation based on ensembles of predictive clustering trees. They evaluate their approach on the ImageCLEF 2007 and 2008 medical image annotation datasets (using IRMA code labels), outperforming both non-hierarchical multi-label classifiers based on support vector machines (SVMs) and single-classifier HMC approaches. Fan et al. [68] propose an HMC classifier for video concept annotation using a concept ontology. They evaluate their approach in the domain of surgery education videos, where concepts are linked to features derived from salient objects [129].

2.6 Multi-View Learning

The effectiveness of machine learning may be improved if training samples are available in multiple, redundant representations called *views*. Examples of multi-view representations include images of the same object taken from different viewing angles, translations of a document into different languages, and visual and textual features of an image and its caption found in scientific articles. Machine learning algorithms for multi-view data have been developed and investigated in the *multi-view learning* research field [202, 233] during the last two decades, and we would like to take advantage of appropriate algorithms to help improve MCR. This is a promising approach due to the fact that medical

[15] http://si.washington.edu/projects/fma
[16] http://www.snomed.org/snomed-ct/
[17] https://www.nlm.nih.gov/research/umls/
[18] https://www.ncbi.nlm.nih.gov/
[19] http://bioportal.bioontology.org/

cases are often available in multiple, complementary representations like textual descriptions, diagnostic images, and annotations taken from a biomedical ontology or controlled vocabulary. However, as machine learning approaches often focus on classification, regression, or clustering problems, only a subset of them will be helpful for an information retrieval problem like MCR.

Xu et al. [233] classify existing multi-view learning approaches into three groups: (1) co-training, (2) multiple kernel learning, and (3) subspace learning methods. *Co-training* has originally been proposed as a method for semi-supervised learning [26], but is naturally applicable to multi-view learning problems. Learning algorithms are trained separately on each view and used to make predictions on unlabeled examples. Examples with high-confidence predictions are then added to the training set and used in an iterative training process. Co-training aims at maximizing the agreement of predictors on different views and will be successful if (among other assumptions) the views are conditionally independent given a class label. Several extensions or modifications of co-training have been proposed to relax its assumptions (like co-EM [153, 30]), transform it to a regularized optimization problem (co-regularization [189]), combine it with graph-based learning methods [240], or apply it to regression [29] or clustering problems [23]. Because co-training is basically a training or optimization strategy that does not produce an obvious relation between input samples that could help multimodal retrieval, we will not consider this class of multi-view learning algorithms for MCR.

Multiple kernel learning (MKL) can be viewed as a strategy to improve the effectiveness of kernel-based machine learning algorithms (like kernel SVM) by automatically learning how to combine or how to select hyper-parameters of multiple kernels. Kernels are typically used to build the disciminant function of classifiers and correspond to different notions of similarity in the feature space. Hence, MKL can naturally be applied to the multi-view setting, where different kernels exploit the specific data distributions of different views. The extensive literature on MKL has been reviewed by Gönen and Alpaydın [74], and in the context of multi-view learning also by Xu et al. [233]. Despite the interpretation of kernel functions as dissimilarity measures, we found no indication in literature that MKL methods have been applied to multimodal retrieval problems.

Subspace learning methods project multiple views into a shared (low-dimensional) latent space by assuming that all views can be generated from latent space. From the viewpoint of MCR, we are interested in two potential uses of these learning methods: multimodal retrieval and multi-label classification (concept mapping, see Chapter 4). We therefore group existing subspace learning methods into (a) approaches supporting multimodal retrieval, (b) methods enabling multi-label classification, and (c) other approaches. Since we consider subspace learning approaches more relevant to MCR than others, we will describe the three groups of existing methods in more detail in the following three subsections.

Note that any retrieval method may also be used for multi-label classification by harvesting labels of retrieved items (implementing a nearest-neighbor classifier), and that multi-label classification may be used to implement multimodal retrieval (concept-based retrieval, see Chapter 6). The grouping of subspace learning methods is therefore based on the primary purpose of the various approaches.

Finally, we note that Sun [202] categorizes existing multi-view learning approaches into two classes only: co-training and co-regularization methods, where most MKL and subspace learning approaches are assigned to the co-regularization category, but also Bayesian co-training [240]. Although Sun does not refer to MKL approaches explicitly, some of his multi-view SVM approaches [203, 201] can be regarded as such.

2.6.1 Subspace Learning for Multimodal Retrieval

Subspace learning methods supporting multimodal retrieval learn a similarity measure or probability distribution in latent space, that can be used to retrieve most similar or most likely instances given a query representation.[20] We identify three classes of existing subspace learning approaches suitable for multimodal retrieval: CCA-based approaches, metric learning methods, and probabilistic latent variable models.

Canonical correlation analysis (CCA) [92] was one of the earliest methods applied to multi-view subspace learning. It computes linear projections to a latent space where the statistical correlation of two views is maximized. To overcome the limitations of linear analysis, several non-linear extensions of CCA have been developed, including kernel CCA [113], SVM-2K [69], and sparse CCA [83]. Like SVM-2K, generalized multi-view analysis [185] is a supervised extension of CCA that additionally tries to separate latent representations of instances belonging to different classes in order to facilitate classification in latent space. The fixed constraint on the dimensionality of latent space has been relaxed by reformulating and generalizing CCA as a convex optimization problem [225].

From the viewpoint of cross-modal retrieval, CCA has been used to improve concept-based retrieval, where multi-class logistic regression is applied to the shared latent space produced by CCA, in order to map text and images into a low-dimensional space of semantic labels [160]. Remarkably, this study suggests that exploiting the correlation between views (as done by CCA) and learning more abstract representations (by multi-label classification) have positive complementary effects on cross-modal retrieval performance and hence can be combined with benefit. Another interesting cross-modal retrieval approach based on CCA is the three-view embedding of representations obtained from explicit feature kernel mappings by Gong et al. [75]. They show that adding

[20] The silent assumption is that most similar or most likely instances are also *most relevant*, i.e. the similarity or probability measure is able to reflect the relevance of instances with respect to a given query.

semantic labels (ground-truth keywords) of internet images as a third view in addition to textual and visual views improves retrieval performance, even when it is generated from other views by supervised or unsupervised methods. We expect that both approaches [160, 75] are applicable to MCR where biomedical concepts (see Chapter 4) take the role of semantic labels.

Metric learning approaches operate in a supervised or semi-supervised setting and aim at learning a metric (dissimilarity measure) in and across multi-view feature spaces such that representations of similar instances (as determined by ground-truth labels) are close to each other with respect to this metric.

Yu et al. [239] extend a graph-based semi-supervised classification scheme named Local and Global Consistency [248] to jointly learn Mahalanobis metrics for single-view representations, their linear combination weights for multi-view representations, and class confidence values for labeled and unlabeled examples by solving a non-convex optimization problem. The resulting similarity measure is applicable to arbitrary multi-view representations, but classification of unseen examples is not directly supported.

Efficient large-scale multimedia retrieval methods based on nearest neighbor search like cover trees [21] or hashing approaches [223, 111] rely on the Euclidean distance of feature representations. It is therefore desirable to embed multi-view representations into a shared latent space where similar instances are close to each other with respect to Euclidean distance. Because such an embedding naturally induces a dissimilarity measure in and across multi-view feature spaces, we call it a *multi-view metric embedding*.

Quadrianto and Lampert [156] propose to learn a multi-view metric embedding by solving an optimization problem that maps different views of similar instances to nearby points in shared latent space, while pushing dissimilar instances apart. The embedding of each view is parameterized as a linear combination of fixed (non-linear) basis functions, and the loss function of the optimization problem is chosen such that it can be decomposed into a difference of two concave functions, allowing an efficient solution by the concave-convex procedure [241].

Zhai et al. [245] go beyond learning a global multi-view metric embedding, which uses the same parameters for mapping all multi-view representations to shared latent space, and propose to determine a locally smooth linear embedding for every unlabeled or previously unseen data point in one of the multi-view feature spaces. The corresponding convex optimization problem has an efficient closed-form solution and uses a globally consistent non-parametric embedding of labeled multi-view presentations learned in the training phase.

A third group of subspace learning methods supporting multimodal retrieval is based on *probabilistic latent variable models*, which are assumed to generate multiple view representations by probabilistic processes. Advantages of these approaches include their flexibility, elegance, capability of learning from small training sets (i.e. generalization

ability), and the inference of conditional probabilities that could be used for ranking in retrieval or for multi-label classification. The major issue for practical applications is still time complexity of both training and inference algorithms, which may prohibit their use with medium-sized or large datasets.

An early approach falling into this category is the Shared Gaussian Process Latent Variable Model (sGPLVM) by Shon et al. [186]. The method learns a probabilistic model of shared latent variables generating two views of training instances, together with radial basis function (RBF) kernels used to map latent points to view representations. Additionally, inverse mappings from view feature spaces to the latent space are learned.

The Shared Kernel Information Embedding (sKIE) approach of Memisevic et al. [137] models probability distributions of data points in view feature spaces and in latent space by kernel density estimates (KDEs), which capture the geometric structure of data points while supporting a probabilistic nonlinear and non-parametric embedding into latent space. Latent representatives used to define the KDE in latent space are learned by maximizing (an approximation of) the mutual information between the multi-view feature distribution and the latent distribution, assuming conditional independence between multiple views given a latent representation. In contrast to the sGPLVM [186], the result of embedding a multi-view representation is not a single latent point, but an explicit conditional probability density over latent space, allowing to deal with ambiguities of the embedding (multimodal density) or generating representative samples. Training has time complexity $O(N^2)$ in the number N of training samples, and inference needs $O(N)$ operations, because KDEs are defined in terms of training samples. Note that training and the search for local maxima in multimodal densities for inference involve gradient-based iterative optimization, which may lead to efficiency problems on large datasets, even with a GPU-based parallelized implementation.[21]

Chen et al. [42] model multi-view and latent representations by undirected Markov networks and estimate parameters by jointly maximizing the data likelihood and minimizing the hinge loss of supervised training data, following the idea of large margin classifiers like SVM. Learning and inference problems are solved approximately with a contrastive divergence method [224], and experiments are conducted on rather small video and image datasets with a small number of class labels. Although the approach could be applied successfully to image classification, retrieval, and annotation with these datasets, the use of larger datasets and inference for unseen examples may lead to practical problems.

[21] At the time of this writing, the URL of the provided implementation announced in [137] was no longer available.

2.6.2 Subspace Learning for Multi-label Classification

Multi-label classification [246] is the problem of assigning one or more predefined class labels to a test instance. Aiming at the potential use of such techniques for biomedical concept mapping (see Chapter 4), we are particularly interested in methods that support a large number (thousands) of class labels and can be applied to previously unseen instances (case queries).

A number of potentially useful approaches *factorize latent space* into a subspace shared by multiple views and subspaces that are private to each view. The objective is to improve the reconstruction ability for a single view given latent representations and to leverage private information of each view for improved classification.

Salzmann et al. [177] proposed a regularization strategy that can be applied to any optimization-based approach to learn factorized orthogonal latent spaces. Regularization terms encourage the pairwise orthogonality between shared and private subspaces, and between private spaces. Additionally, a trace norm regularizer encourages a low rank representation of training samples in each view-specific latent space, and by encouraging conservation of spectral energy in both original view and latent representations trivial solutions are avoided. The approach is applied to probabilistic latent variable models [186, 137] (see Section 2.6.1) and delivers improved results in a human pose estimation experiment, which represents a cross-modal regression problem.

The idea of factorized latent spaces has been further pursued in two different directions. Jia et al. [101] employed structured sparse coding techniques to learn view-dependent dictionaries that employ only a subset of latent dimensions, and at the same time discover the dimensionality of the latent space while encouraging a low-dimensional shared subspace. Compared to [177], the approach allows for more efficient (iterative convex) optimization and delivers better results for human pose estimation. As sparse coding can be seen as a specific method of representation learning [19], Ye et al. [238] used another popular representation learning algorithm, namely a neural network with a single hidden layer in autoencoder configuration, to learn shared and private latent variables that are able to generate the given views. Private latent spaces are guaranteed to be orthogonal by construction of the neural network, and the robustness of the learning algorithm is improved by introducing noise into training data, following the strategy of denoising autoencoders [216]. Promising results were obtained on the PASCAL VOC2007 dataset (10k images) for multi-label object classification, but hyper-parameter optimization (number of hidden neurons, found by random search [20]) seems to be a major issue for practical applications.

Recently, *coupled dictionary learning* [94] has been combined with *coupled feature selection* [220] for cross-modal multimedia retrieval [235, 236]. These approaches are supervised in the sense that they learn projections of two views into a space of semantic ground-truth labels. The projections are jointly optimized by L_{21} norm regularization

to encourage the selection of relevant and discriminative features [87, 152], and by trace norm regularization to impose a low-rank constraint on the correlated embedding of the two views into label space [79, 82]. While Wang et al. [220] learn linear projections directly from feature representations of the two views, Xu et al. [235] first perform coupled dictionary learning to obtain sparse and homogeneous representations of the views, which are then used to learn projections into label space. We consider this approach as another good candidate for MCR, because it can be used directly to map medical cases (consisting of textual and visual representations) to a low-rank linear combination of biomedical concepts, enabling concept-based retrieval (see Chapter 6). Compared to the three-view CCA approach of Gong et al. [75] (see Section 2.6.1), coupled dictionary learning and feature selection [235] seems to be advantageous for MCR, because it requires less design and parameter choices and does not need explicit dimensionality reduction of raw feature representations.

2.6.3 Other Subspace Learning Approaches

A class of subspace learning methods that cannot easily be applied to multimodal retrieval or multi-label classification emerged from *nonlinear dimensionality reduction* techniques producing topology-preserving (smooth) embeddings [118, Chapter 5]. Some of them are based on spectral methods [179], which find low-dimensional representations by using eigenvectors of specially constructed matrices, like multi-view spectral embedding [231] and a similar approach combining it with sparse coding [81]. Multi-view stochastic neighbor embedding [232] defines a probability distribution on sample pairs encoding their distances in the original feature spaces and a corresponding distribution in shared latent space, and finds latent representations by minimizing the Kullback-Leibler divergence between the two distributions. In general, these techniques cannot be easily utilized for MCR, because they do not produce a parametric embedding that could be applied to previously unseen instances; they are designed to reduce the dimensionality of a given multi-view dataset only.

Diethe et al. [62] generalized Fisher discriminant analysis to the multi-view case, which learns a projection direction for every view that minimizes variance of data along its direction while maximizing the distance between the average outputs for each class. Kernel and convex formulations of the optimization problem are provided, including sparse regularizers, but analogously to SVM, the application to multi-label classification problems is cumbersome.

Chen et al. [43] treat cross-modal retrieval as a regression problem and propose to apply continuum regression methods [28] to find latent view-specific subspaces with maximal covariance whose correlation is modeled by linear regression. Partial least squares methods [169] can be seen as special cases of continuum regression and are applied to a small dataset of Wikipedia documents with rather low-dimensional view

representations (10-topic LDA text representation and 128-codeword SIFT image representation). Cross-modal retrieval results compare well to the CCA-based approach of Rasiwasia et al. [160] (see Section 2.6.1). Note, however, that this approach cannot be easily applied to multimodal retrieval or multi-label classification problems.

CHAPTER 3

Biomedical Articles and Images

This chapter introduces the dataset of scientific biomedical articles representing a collection of medical case descriptions that are used for experiments. In addition, the preprocessing of article images (figures) intended to support retrieval and concept mapping of images (Chapter 4) is explained. As we proposed a novel algorithm for separation of compound figures [210], we also include experimental results of its evaluation.

3.1 Biomedical Article Dataset

The main dataset used for most of the experiments in this thesis consists of about 75,000 scientific biomedical articles in English language used for ImageCLEF medical tasks since 2012 [104], referred to as *ImageCLEF MCR dataset* throughout this thesis. Those articles were retrieved from PubMed Central[1] by selecting open access journals that allow for free redistribution of data. The articles of this dataset contain about 300,000 images of unconstrained modalities (biomedical images, diagrams, charts, photographs, etc.). To evaluate retrieval performance, 35 queries representing patients' symptom descriptions and corresponding diagnostic images are available, together with relevance judgments for a limited number of articles in the dataset.

Some dataset statistics are presented in Table 3.1. The number of indexed terms was determined after indexing the document collection using Lucene with its default token analyzer, which performs stop word removal and stemming. The number of relevance judgments corresponds to the number of articles that have been judged by medical experts whether they are relevant for a given query or not. Three of the queries in the dataset have only one relevant article according to relevance judgments.

Fig. 3.1 depicts the distribution of images per article for this dataset. Interestingly, there is only one article without any images, and the distribution attains a maximum for one image per article.

[1] `http://www.ncbi.nlm.nih.gov/pmc/`

Table 3.1: Summary statistics of the MCR dataset.

	Total	*Min*	*Max*	*Average*
Articles	74,654			
Indexed terms	490,273			
Document length (terms)		50	43,524	3,479
Images	306,549			
Images per article		0	71	4.1
Queries	35			
Query length (terms)		16	47	30.9
Images per query		2	3	2.2
Relevance judgments	15,028			
Judged articles per query		372	480	429
Relevant articles per query		1	101	20.3

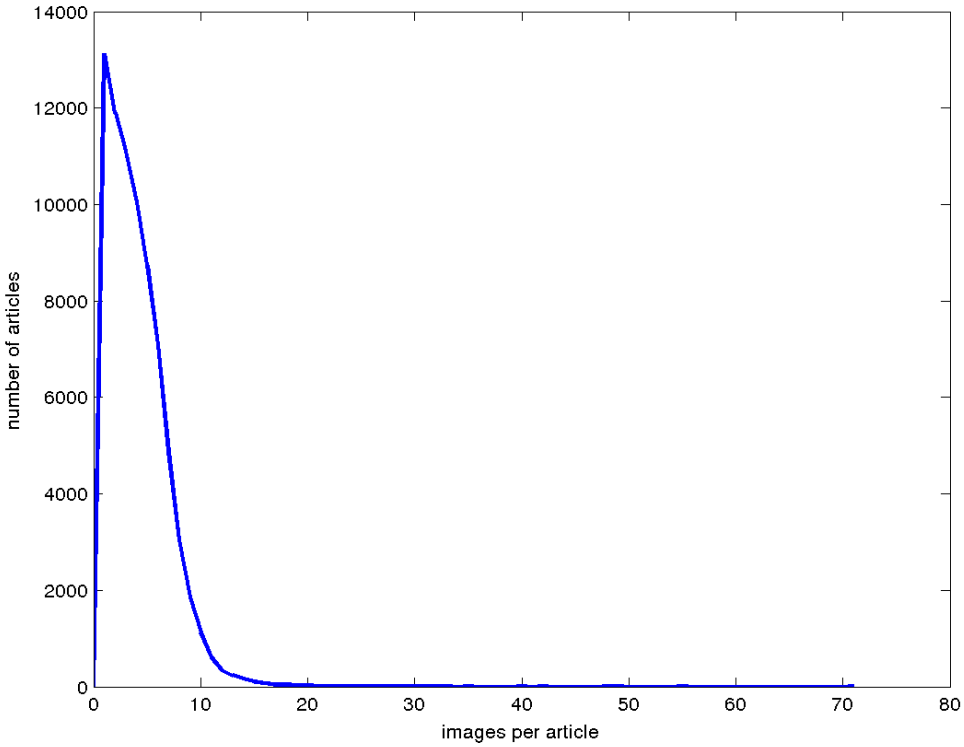
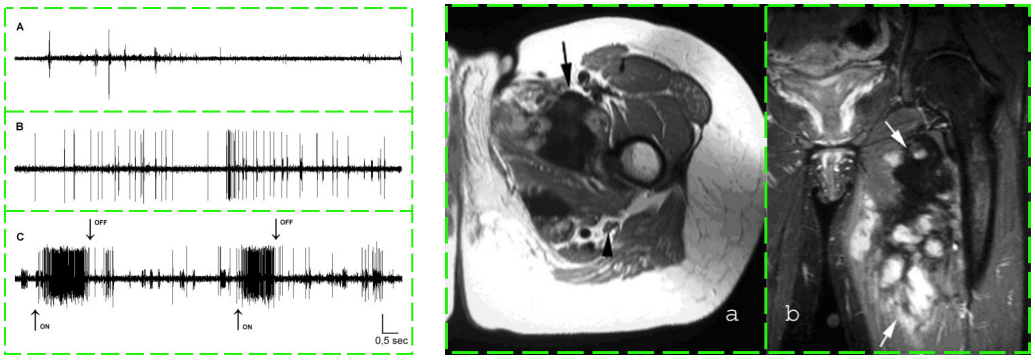


Figure 3.1: Distribution of images per article in the ImageCLEF MCR dataset.



(a) (b)

Figure 3.2: Sample compound images of the ImageCLEF MCR dataset suitable for two different separator detection algorithms. Subfigures are separated by (a) whitespace, (b) a vertical edge. Dashed lines represent the expected output of CFS.

Articles are available as XML documents with separate fields for title, authors, abstract, fulltext, and figure captions (see Fig. 1.1 on page 2). Every image occurring in articles is equipped with a caption text. Image files are stored separately (most of them in JPEG format) and can be associated with articles using their figure ID.

3.2 Article Image Preprocessing

Articles in scientific publications contain a substantial amount of figures consisting of two or more subfigures, which could be treated as separate images for the purpose of automatic content-based analysis or indexing for retrieval. Figure 3.2 shows two examples of such *compound figures* found in the ImageCLEF MCR dataset. Based on published datasets drawn from open access biomedical literature, it has been estimated that between 40% and 60% of figures occurring in articles are compound figures [7, 44, 88].

Compound figures hamper content-based analysis and indexing of article images for retrieval, because global image features extracted from a compound image are a mixture (often an average) of the same features extracted from the subfigures only, leading to reduced discriminative power of these features on compound images. The situation may be slightly better for local image features, which capture the existence of certain texture or shape patterns in small image regions, but the predominant way of aggregating local features of an image in a Bag of Visual Words representation [190] still suffers from the additive effect of including local features from all subfigures. Moreover, subfigures of a given compound image usually convey different semantic information that may be relevant for retrieval, although the compound figure establishes a common semantic context for subfigures.

It is therefore desirable to automatically recognize and separate compound figures before using them for medical case retrieval. Since research on this subject is rather young [88] and corresponding tools are not publicly available, we devised and evaluated an approach for *compound figure classification* (CFC) and *compound figure separation* (CFS) in previous work [208, 209, 210], described in the following sections.

CFC (Section 3.2.1) is a binary classification problem that aims at discriminating between compound and non-compound figures given an article image. CFS (Section 3.2.2) is the problem of determining the bounding boxes of all subfigures of a given compound figure. Algorithms solving the CFC and CFS problems are naturally combined into a *CFC-CFS process chain* (Section 3.2.3) that receives arbitrary article images as input and delivers bounding boxes of subfigures (or of single figures) at the output. Images classified as *compound* by the CFC algorithm are further processed by CFS, whereas for images predicted as *non-compound* by CFC a bounding box covering the entire image is produced.

3.2.1 Compound Figure Classification

Recognizing compound figures in a dataset of article images can be viewed as a binary classification problem. We address this problem by using hand-crafted image features and classical machine learning algorithms, because we consider the available training datasets as being too small for deep learning techniques [18, 19], and we expect that the effect of limited classification accuracy on the CFC-CFS process chain can be partly compensated by biasing the classifier towards the *compound* class (see Section 3.2.3).

For CFC, we propose to use three types of image features determined separately for vertical and horizontal directions of a gray-scale image whose pixel values have been normalized to the range $[0, 1]$. Each feature type is computed by aggregating each pixel line in direction D (vertical or horizontal) to a single real number, resulting in a single *projection vector* representing the image along direction D' orthogonal to D. The spatial distribution of values in the projection vector is then captured by a *spatial profile vector* of fixed length. The final feature vector is formed by concatenating the horizontal profile vectors of the three feature types, followed by the corresponding vertical profile vectors.

The three feature types differ in how the projection vector is calculated: (1) *mean* gray values along pixel lines, (2) *variance* of gray values along pixel lines, and (3) one-dimensional *Hough transform*, which counts the number of edge points aligned in direction D in a binary edge map of the input image. The binary edge map is produced by applying a gradient threshold on edges in direction D detected by the Sobel operator. Hough transform values are then normalized to the range $[0, 1]$ using the image dimension in direction D (width or height). Some of the spatial profile methods applied afterwards require *quantization* of projection vectors, which is performed differently for the three feature types, using quantization parameters (positive integers) p, q, and h:

- Mean projection values are quantized into p bins dividing $[0, 1]$ into p subintervals with lower bounds $1 - 2^{i-p}$ for $i = 1, 2, \ldots, p$. The logarithmic scale for quantization should help to discriminate between high values (white separator bands) and others.

- Variance projection values are quantized into q bins dividing $[0, 1]$ into q subintervals with upper bounds 2^{i-q} for $i = 1, 2, \ldots, q$. The logarithmic scale for quantization should help to discriminate between low-variance pixel lines (subfigure separators) and others.

- Normalized Hough transform values are quantized into h bins dividing $[0, 1]$ into h subintervals with lower bounds $1 - 2^{i-h}$ for $i = 1, 2, \ldots, h$. The logarithmic scale for quantization should help to discriminate between Hough peaks (subfigure separators) and others.

We consider six spatial profile methods to produce profile vectors from projection vectors. Five of them require quantization of projection vectors and divide the vector of dimensionality N into k *spatial bins* of $\lfloor N/k \rfloor$ or $\lfloor N/k \rfloor + 1$ adjacent positions. An additional profile method tries to capture the spatial structure of the projection vector using its Fast Fourier Transform (FFT).

- *Profile 1:* A spatial bin is represented by the full normalized histogram of quantized projection values, resulting in p, q, or h values per spatial bin.

- *Profile 2:* A spatial bin is represented by the quantized projection value that occurs most often (the mode). This value is then normalized to the range $[0, 1]$.

- *Profile 3:* A spatial bin is represented by the relative frequency of the largest quantized projection value, resulting in a single number in the range $[0, 1]$.

- *Profile 4:* A spatial bin is represented by its maximum quantized projection value, normalized to the range $[0, 1]$.

- *Profile 5:* A spatial bin is represented by its average quantized projection value, normalized to the range $[0, 1]$.

- *Profile 6:* The absolute values of the first k low-frequency FFT coefficients of the projection vector are normalized by $1/N$, such that resulting values are constrained to the range $[0, 1]$.

The dimensionality of feature vectors depends on parameters k, p, q, h, and on the profile method used for each of the three feature types, as presented in Table 3.2. We denote a certain *feature set* by three numbers xyz representing the spatial profile

Table 3.2: Dimensionality of various feature sets used for compound figure classification. k denotes the number of spatial bins used to compute profile vectors. p, q, and h are quantization parameters. The right-most column gives the dimensionality for parameter settings $k = 16$, $p = 5$, $q = 8$, $h = 3$.

Feature Set	*Dimensionality*	*Example*
111	$2 * k * (p + q + h)$	512
222	$6 * k$	96
333	$6 * k$	96
444	$6 * k$	96
555	$6 * k$	96
666	$6 * k$	96
011	$2 * k * (q + h)$	352
034	$4 * k$	64
134	$2 * k * (p + 2)$	224
434	$6 * k$	96

numbers of mean projection (x), variance projection (y), and Hough Transform (z). A value of zero (e.g. $x = 0$) means that the corresponding component of the feature vector has been dropped. For example, the feature set 034 denotes a feature vector formed by concatenation of horizontal profiles of variance projection and Hough Transform, followed by corresponding vertical profiles. Both profile methods (3 and 4) represent a spatial bin by a single number, resulting in k numbers per profile vector, $2k$ numbers for both horizontal profiles, and $4k$ numbers for the final feature vector.

As classifier algorithms we use logistic regression, a linear support vector machine (SVM), and a non-linear SVM with a radial basis function kernel.

3.2.2 Compound Figure Separation

For CFS, we designed an image processing algorithm comprising distinct modules for detecting two types of separators between subfigures: (1) homogeneous rectangular areas of whitespace spanning the entire image width or height, which we call *separator bands* (shown in Fig. 3.2(a)); and (2) *separator edges* spanning the entire image width or height, which may arise from borders drawn around subfigures or from adjacent subfigures "stitched together" as shown in Fig. 3.2(b). The proposed CFS algorithm internally uses a separate binary classifier (independent from CFC) to decide which of the two separator detection modules to apply to a given compound image. Based on the observation that compound images containing graphical illustrations (such as diagrams and charts) often contain separator bands, whereas most subfigures in other compound images show

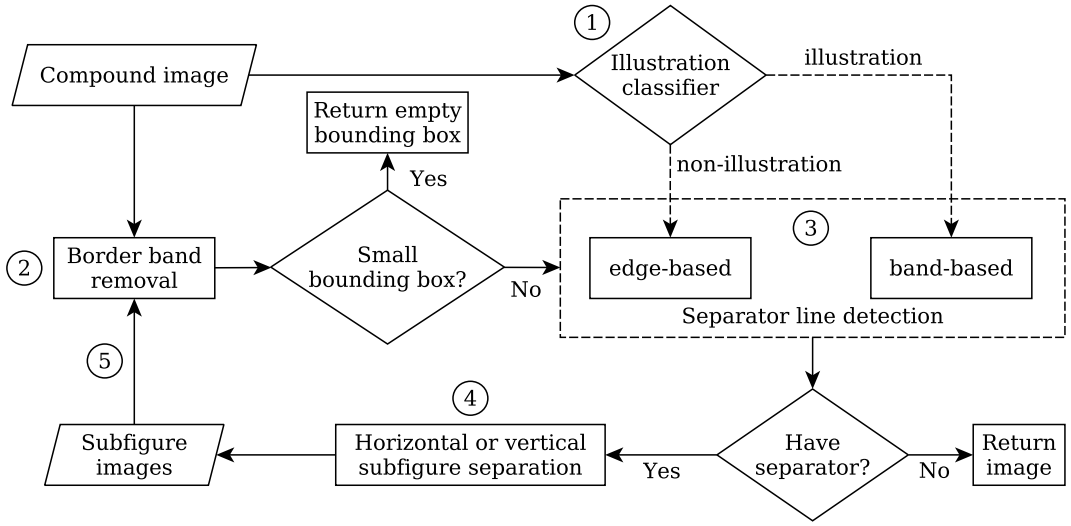


Figure 3.3: Recursive algorithm for compound figure separation. Numbers denote the main algorithmic steps described in Section 3.2.2.

rectangular border edges, we train the internal CFS classifier to discriminate between graphical illustrations and other article images and call it *illustration classifier*.

Figure 3.3 presents the proposed recursive algorithm for CFS comprising the following steps: (1) classification of the compound image as illustration or non-illustration image, (2) removal of border bands, (3) detection of separator lines, (4) vertical or horizontal separation, and (5) recursive application to each subfigure image. The *illustration classifier* is used to decide which of two separator line detection modules to apply: if the compound image is classified as an illustration image, the *band-based* algorithm is applied, which aims at detecting separator bands between subfigures. Otherwise, the image is processed by the *edge-based* separator detection algorithm, which applies edge detection and Hough transform to locate candidate separator edges. The algorithm selection is based on the assumption that edge-based separator detection is better suited for non-illustration compound images due to visible vertical or horizontal edges separating subfigures. Note that this assumption is not violated by non-illustration compound images with separator bands where subfigures have a visible rectangular border. The following four sections describe the illustration classifier, the main recursive algorithm, and the two separator detection modules in more detail.

3.2.2.1 Illustration Classifier

The illustration classifier is used to decide which separator detection algorithm to apply to a given compound image. If the image is predicted to be a graphical illustration with

probability greater than `decision_threshold`, the band-based separator detection is applied, otherwise the edge-based separator module is used. This decision is made only once for each compound image, so all recursive invocations use the same separator detection algorithm.

Due to promising effectiveness for CFS in early experiments, we use four sets of global image features as classifier input, computed after gray-level conversion: (1) *simple2* is a two-dimensional feature consisting of image entropy, estimated using a 256-bin histogram, and mean intensity; (2) *simple11* extends *simple2* by 9 quantiles of the intensity distribution; (3) *CEDD* is the well-known color and edge directivity descriptor [40] (144-dimensional); and (4) *CEDD_simple11* is the concatenation of *CEDD* and *simple11* features (155-dimensional).

As machine learning algorithms we consider support vector machines (SVM) with radial basis function kernel (RBF) and logistic regression. Although logistic regression is generally inferior to kernel SVM due to its linear decision boundary, it has the advantage of providing prediction probabilities, which allow us to tune the selection of separator detection algorithms using the `decision_threshold` parameter.

3.2.2.2 Recursive Algorithm

Before applying the main algorithm (Fig. 3.3) to a given compound figure image, it is converted to 8-bit gray-scale. *Border band removal* detects a rectangular bounding box surrounded by a maximal homogeneous image region adjacent to image borders (border band). If the resulting bounding box is empty or smaller than `elim_area` or if maximal recursion depth has been reached, an empty bounding box is returned, terminating recursion. The *separator line detection* modules are invoked separately for vertical and horizontal directions, so they deal with a single direction θ and return a list of corresponding separator lines. An empty list is returned if the respective image dimension (width or height) is smaller than `mindim` or if no separator lines are found. If the returned lists for both directions are empty, recursion is terminated and the current image (without border bands) is returned. The *decision about vertical or horizontal separation* is trivial if one of both lists of separator lines is empty. Otherwise the decision is made based on the regularity of separator distances: locations of separator lines and borders are normalized to the range [0,1], and the direction (vertical or horizontal) yielding the lower variance of adjacent distances is chosen. Finally, the current figure image is divided into subimages along the chosen separation lines, and the algorithm is applied recursively to each subimage.

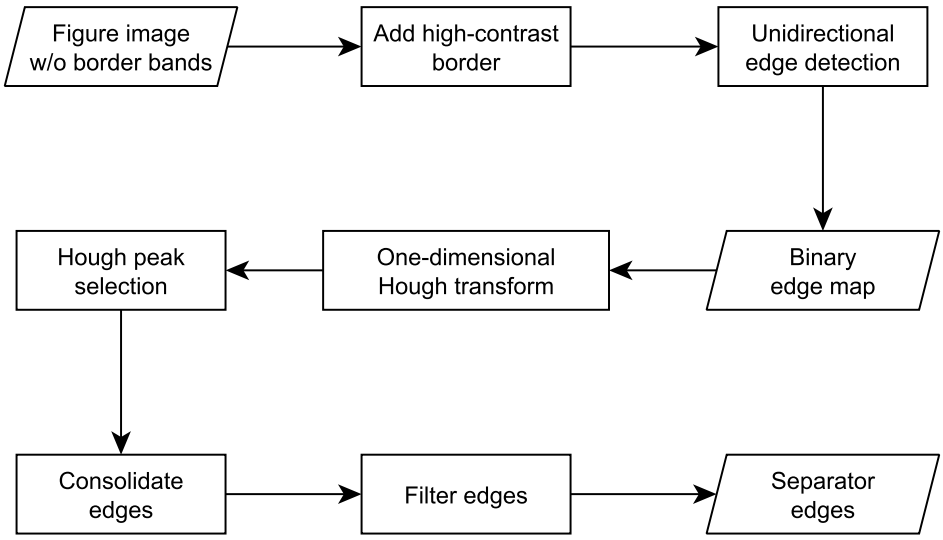


Figure 3.4: Edge-based separator line detection.

3.2.2.3 Edge-based Separator Detection

The edge-based separator line detection algorithm aims at detecting full-length edges of a certain direction θ (vertical or horizontal) in a given gray-scale image. It comprises the following processing steps depicted in Fig. 3.4: (1) unidirectional edge detection, (2) peak selection in one-dimensional Hough transform, and (3) consolidation and filtering of candidate edges.

Edge detection is implemented by a one-dimensional Sobel filter und subsequent thresholding (`edge_sobelthresh`) to produce a binary edge map. The one-dimensional Hough transform counts the number of edge points aligned on each line in direction θ. So the peaks correspond to the longest edges, and their locations identify candidate separator edges. To make borders appear as strong Hough peaks, we add an artificial high-contrast border to the image prior to edge detection. Peaks are identified by an adaptive threshold t that depends on the recursion depth k (zero-based), the maximal value m of the current Hough transform, and the fill ratio f of the binary edge map (fraction of non-zero pixels, $0 \leq f \leq 1$), see (Eq. 3.1). α and β are internal parameters (`edge_houghratio_min` and `edge_houghratio_base`).

$$h = \alpha * \beta^k \ , \qquad t = m * \left(h + (1-h) * \sqrt{f}\right) \ . \tag{3.1}$$

The rationale behind these formulas is to cope with noise in the Hough transform. Hough peaks were observed to become less pronounced as image size decreases (implied by increasing recursion depth) and as the fill ratio f increases (more edge points increase the

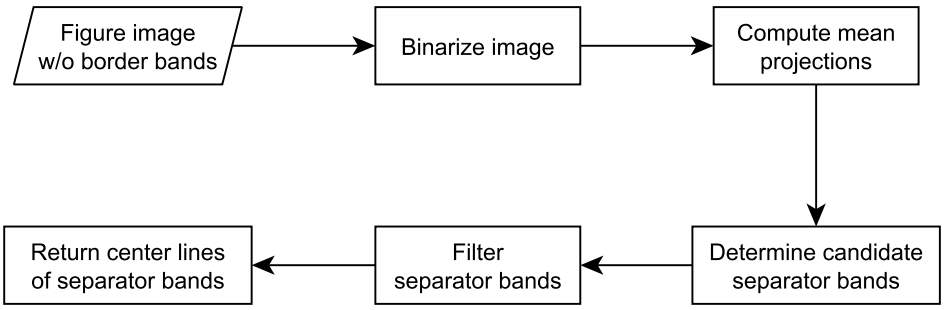


Figure 3.5: Band-based separator line detection.

probability that they are aligned by chance). Equation (3.1) ensures a higher threshold in these cases. Additionally, as recursion depth increases, the algorithm should detect only more pronounced separator edges, because further figure subdivisions become less likely.

Hough peak selection also includes a similar regularity criterion as used for deciding about vertical or horizontal separation (see Section 3.2.2.2): the list of candidate peaks is sorted by their Hough values in descending order, and candidates are removed from the end of the list until the variance of normalized edge distances of remaining candidates falls below a threshold (`edge_maxdistvar`). Candidate edges resulting from Hough peak selection are then consolidated by filling small gaps (of maximal length given by `edge_gapratio`) between edge line segments (of minimal length given by `edge_lenratio`). Finally, edges that are too short in comparison to image height or width (threshold `edge_minseplength`), or too close to borders (threshold `edge_minborderdist`) are discarded.

3.2.2.4 Band-based Separator Detection

The band-based separator detection algorithm aims at locating homogeneous rectangular areas covering the full width or height of the image, which we call *separator bands*. Since this algorithm is intended primarily for gray-scale illustration images with light background, we assume that separator bands are white or light gray. The algorithm consists of four steps illustrated in Fig. 3.5: (1) image binarization, (2) computation of mean projections, (3) identification and (4) filtering of candidate separator bands.

Initially, we binarize the image using the mean intensity value as a threshold. We then compute mean projections along direction θ (vertical or horizontal), that is, the mean value of each line of pixels in this direction. A resulting mean value will be 1 (white) if and only if the corresponding line contains only white pixels. Candidate separator bands are then determined by identifying maximal runs of ones in the vector

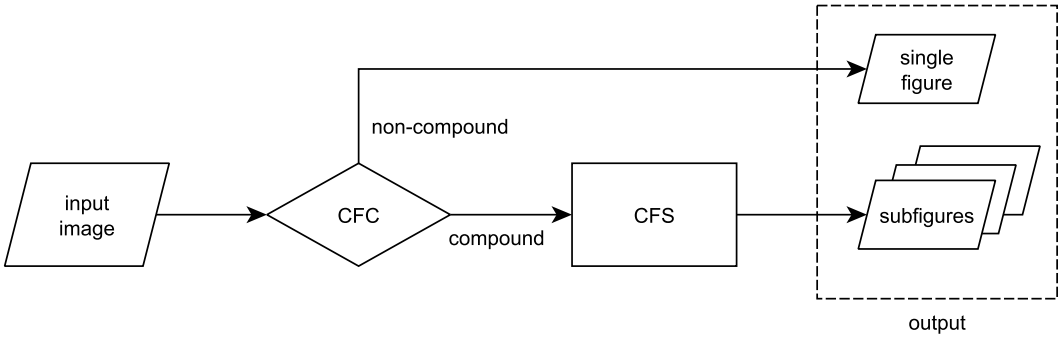


Figure 3.6: Process chain consisting of compound figure classifier (CFC) and compound figure separation (CFS).

of mean values that respect a minimal width threshold (`band_minsepwidth`). They are subsequently filtered using a regularity criterion similar to Hough peak selection (see Section 3.2.2.3), this time using distance variance threshold `band_maxdistvar`. Finally, selected bands that are close to the image border (threshold `band_minborderdist`) are discarded, and the center lines of remaining bands are returned as separator lines.

3.2.3 CFC-CFS Chain

Processing compound figures in a collection of scientific articles is expected to happen in a two-stage process as illustrated in Fig. 3.6: (1) all article images are classified as *compound* or *non-compound* by applying a compound figure classifier (CFC); (2) the predicted *compound* images are then processed by a compound figure separation (CFS) algorithm to obtain subfigures. The resulting set of subfigures and predicted *non-compound* figures can then be used for further application-specific processing (e.g. content-based indexing for retrieval). We are therefore interested in evaluating and improving the effectiveness of the *CFC-CFS process chain*, i.e. the quality of obtained subfigures and non-compound figures with respect to a gold standard and evaluation procedure (see Section 3.3).

A guiding principle for improving the CFC-CFS chain is derived from consideration of the loss of effectiveness caused by different types of CFC errors: *false negatives* (compound figures classified as non-compound) may result in a larger loss than the same number of *false positives* (non-compound figures classified as compound), because false negatives are not processed by CFS and hence all contribute to the loss of effectiveness. On the other hand, there is a chance that false positives are not divided into subfigures by CFS (because it does not detect separation lines), and such instances of false positives will not degrade effectiveness of the CFC-CFS chain. Effectiveness can therefore be optimized on a validation set by biasing CFC decisions towards the *compound* class.

However, this is easy to achieve only for CFC algorithms that deliver predicted class probabilities, like logistic regression, but not for SVM.

The different importance of misclassifications of a binary classifier depending on true classes can be expressed by a 2×2 misclassification loss matrix (Eq. (3.2)) [24]. Rows correspond to true classes and columns to predicted classes, where in the case of CFC the first row or column is assigned to class *non-compound* (C_0) and the second row or column to class *compound* (C_1). The entries of loss matrix (Eq. (3.2)) denote the fact that misclassification of true compound figures incurs a loss that is by a factor of α larger than that of misclassification of true non-compound figures (if $\alpha > 1$). If the classifier is able to predict conditional class probabilities $p(C_k|x)$ for a given image x, the decision of the classifier can be optimized with respect to expected misclassification loss $E_k(x)$ (Eq. (3.3)): image x is assigned to class C_k that minimizes $E_k(x)$ ($k = 0$ or $k = 1$). For the special form of loss matrix given in (Eq. (3.2)), this criterion reduces to a simple threshold on conditional class probability $p(C_1|x)$: image x is assigned to class C_1 if and only if Eq. (3.4) holds. The parameter α can be selected by optimizing effectiveness of the CFC-CFS process chain on a validation set.

$$L = \begin{pmatrix} 0 & 1 \\ \alpha & 0 \end{pmatrix} \tag{3.2}$$

$$E_k(x) = \sum_i L_{ik}\, p(C_i|x) \tag{3.3}$$

$$p(C_1|x) \geq \frac{1}{1+\alpha} \tag{3.4}$$

3.3 Experiments

Experiments were conducted for two different purposes: first, the effectiveness of our proposed CFC-CFS process chain was evaluated and compared to other state-of-the-art algorithms (Section 3.3.3); second, we applied compound figure classification and separation to the ImageCLEF MCR dataset to prepare further use of article images throughout this thesis (Section 3.3.4). Since ground-truth data for compound figures of the MCR dataset is not available, the second type of experimental results cannot be evaluated, but only checked for consistency with compound figure rates reported in the literature.

We evaluate our approach on separate datasets for CFC, CFS, and the CFC-CFS process chain, which are described in Section 3.3.1. As there is no agreement on a standard evaluation protocol for CFS in the research community yet, we use two different evaluation procedures, described in Section 3.3.2. Additionally, we propose to slightly extend existing CFS evaluation protocols in order to apply them to CFC-CFS chains.

Table 3.3: Datasets used in our CFC-CFS experiments. CFC = compound figure classification, CFS = compound figure separation, MC = modality classification; CO = compound, ILL = illustration.

Dataset	*Training*		*Test*	
	Images	Annotations	Images	Annotations
ImageCLEF CFC	10387	6121 CO (59%)	10434	6144 CO (59%)
ImageCLEF CFS	3403	14531 subfigures	3381	12789 subfigures
NLM CFS			380	1656 subfigures
ImageCLEF MC first	1071	607 ILL (57%)	497	261 ILL (53%)
ImageCLEF MC majority	895	514 ILL (57%)	428	243 ILL (57%)
ImageCLEF MC unanimous	867	508 ILL (59%)	398	226 ILL (57%)
ImageCLEF MC greedy	1071	712 ILL (66%)	497	325 ILL (65%)
CFC-CFS	6806	17934 subfigures	6752	16154 subfigures

3.3.1 Datasets

We used several datasets to train and evaluate the different components of our approach in our experiments. All of them were derived from the ImageCLEF MCR dataset (see Section 3.1). A subset of about 21,000 images used for the ImageCLEF 2015 medical tasks [90] formed the basis for most datasets used in our experiments, namely all datasets labeled *ImageCLEF* in Table 3.3.

The CFC training dataset provided by ImageCLEF task organizers contained some erroneous samples (23 images had contradicting annotations), which have been removed from the training set. Table 3.3 refers to the cleaned CFC training set only. The CFC dataset consists of 59% compound images (CO), both in training and test subsets, providing reasonable conditions for training and evaluating a binary classifier.

A similar split of classes is present in the modality classification (MC) datasets, which are used to train and evaluate the binary classifier for illustrations (ILL) (see Section 3.2.2.1). The MC datasets were derived from the dataset of the ImageCLEF 2015 multi-label image classification task [90]. The images are provided with one or more labels of 29 classes (organized in a class hierarchy), which have been mapped to two meta classes: the *illustration* meta class comprises all “general biomedical illustration" classes except for chromatography images, screenshots, and non-clinical photos. These classes and all classes of diagnostic images have been assigned to the *non-illustration* meta class. About 36% of the images in the training set are labeled with multiple classes, corresponding to compound images. Training and evaluation of the illustration classifier (Section 3.2.2.1) requires mapping the set of labels of a given image to a single meta class. We implemented four mapping strategies that first assign each image label to the *illustration* or *non-illustration* meta class, and then operate differently on the list L of meta labels associated with a given image: (1) the *first* strategy simply assigns the first meta label of L to the image; (2) the *majority* strategy selects the meta label

occurring most often in L, dropping the image from the dataset if both meta labels occur equally often; (3) the *unanimous* strategy only assigns a meta label to the image if all meta labels in L are equal, otherwise the image is dropped from the dataset; and (4) the *greedy* strategy maps an image to the *illustration* label if L contains at least one such meta label, otherwise the image is assigned the *non-illustration* label. Note that *majority* and *unanimous* strategies discarded up to 20% of images in the original dataset. Whereas *majority* and *unanimous* mapping strategies are expected to improve classification accuracy, the *greedy* strategy aims at increasing CFS effectiveness based on the assumption that a compound image containing an illustration subfigure is more likely to have separator bands than separator edges.

A research group at the U.S. National Library of Medicine (NLM) had created a dataset to evaluate their CFS approach (and related algorithms) [7] well before the first CFS task at ImageCLEF was issued in 2013. This dataset contains 400 images and 1764 ground-truth subfigures and hence is substantially smaller than the ImageCLEF CFS test dataset. Moreover, it shares 20 images with the training set and 27 images with the test set of the ImageCLEF CFS dataset. The reason for the non-empty intersection of these datasets is that the NLM dataset was sampled from a set of 231,000 article images used at ImageCLEF 2011, which was extended later to the ImageCLEF MCR dataset. Since we used the ImageCLEF CFS training set for parameter optimization, we removed the 20 images in the intersection from the NLM dataset for our experiments. The resulting reduced dataset is listed in Table 3.3 as *NLM CFS* dataset.

For evaluation of the CFC-CFS process chain, we extended the ImageCLEF CFS test dataset (3381 images) with the same number of non-compound images sampled at random from the ImageCLEF CFC test dataset. After removing five images that occurred in both portions of this dataset[2], a test dataset with 6752 images was obtained. In a similar manner, a validation set of 6806 images was constructed from ImageCLEF CFS and CFC training datasets (appearing as "training set" in the last line of Table 3.3). Non-compound images of the CFC-CFS dataset were annotated with a single subfigure covering the entire image, as explained in Section 3.3.2.

3.3.2 Evaluation Methods

While evaluation of classification algorithms is a well-studied problem [143, 36, 85, 192, 108], evaluation of compound figure separation has been addressed by two different ad-hoc procedures only [7, 88]. Both evaluation procedures first determine which detected

[2]Ideally, the intersection should be empty, because the CFS dataset should contain only compound images. However, manual inspection of images in the intersection revealed that both CFS and CFC datasets contain errors and that the distinction between compound and non-compound images is not always clear.

subfigures of a given compound image are correct (*true positive*) with respect to ground-truth subfigures, and then compute an evaluation measure from the number of true positive subfigures over the dataset. However, the way by which true positive subfigures are determined, and which evaluation measures are calculated, differs between the two proposed evaluation procedures, which are described in the sequel. A description of our proposed method to measure the effectiveness of the CFC-CFS process chain completes this section.

3.3.2.1 CFS Evaluation

To describe the evaluation protocols in detail, we introduce the following notation. Without loss of generality, we assume that a subfigure is represented by rectangular area R (bounding box) within an image, and denote its area size (number of contained pixels) by $|R|$. For a given compound figure, let $\{G_i \,|\, i \in I\}$ be the set of ground-truth subfigures, and $\{F_j \,|\, j \in J\}$ the set of subfigures detected by the CFS algorithm that should be evaluated. Note that the overlap area $G_i \cap F_j$ between subfigures is again a rectangle (or empty). The two evaluation protocols employ different definitions of the *overlap ratio* between G_i and F_j, given in Equations (3.5) and (3.6). ρ_{ij}^G is the overlap ratio with respect to ground-truth subfigure G_i, ρ_{ij}^F calculates the ratio with respect to detected subfigure F_j.

$$\rho_{ij}^G = \frac{|G_i \cap F_j|}{|G_i|} \tag{3.5}$$

$$\rho_{ij}^F = \frac{|G_i \cap F_j|}{|F_j|} \tag{3.6}$$

The evaluation procedure used for ImageCLEF CFS tasks [88] iterates over ground-truth subfigures G_i and, for a given G_i, looks for a detected subfigure F_j with maximal overlap ρ_{ij}^F. F_j is associated with G_i if $\rho_{ij}^F > 2/3$ and if F_j has not already been associated with a different ground-truth subfigure. The result is a set of one-to-one associations between ground-truth subfigures and detected subfigures, which are regarded as true positives. Note that although the set of associations may depend on the order of iterations over G_i, the number C of these associations does not. Accuracy can therefore be defined per compound figure as $C/\max(N_G, N_D)$, where N_G and N_D are the numbers of ground-truth and detected subfigures, respectively. Accuracy on the test set is the average of accuracy values computed for each compound figure.

The authors of the NLM CFS dataset [7] (see Section 3.3.1) used a different criterion to determine true positive subfigures. A detected subfigure F_j is considered true positive if and only if there is a ground-truth subfigure G_i with $\rho_{ij}^G > 0.75$ and $\rho_{kj}^G < 0.05$ for all other ground-truth subfigures G_k. That is, subfigure F_j has a notable overlap with one

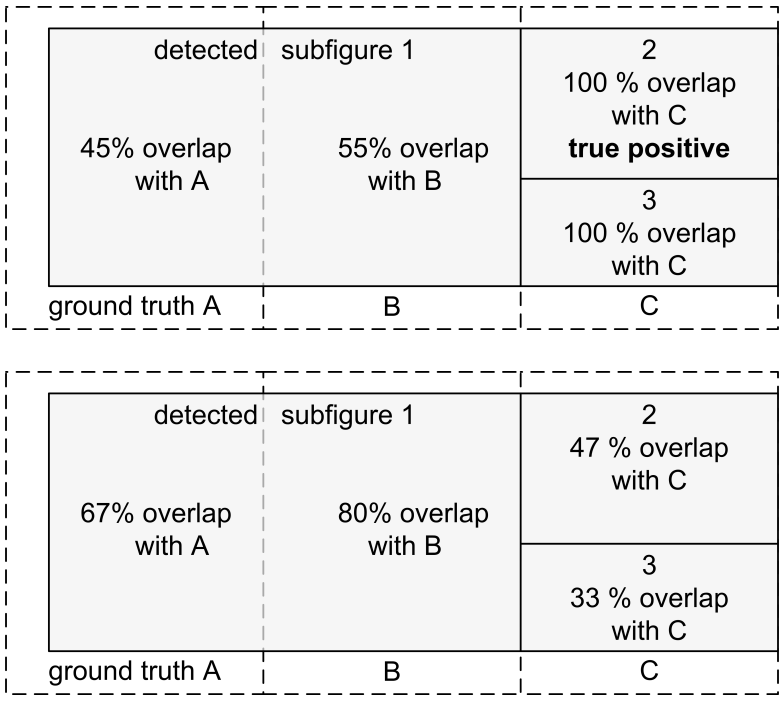


Figure 3.7: Determination of true positive detected subfigures by (a) ImageCLEF and (b) NLM CFS evaluation procedures.

ground-truth subfigure only. Given the total number N of ground-truth subfigures in the dataset, the total number D of detected subfigures, and the number T of detected true positive subfigures, the usual definitions for classifier evaluation measures can be applied to obtain precision P, recall R, and F_1 measure, see Eq. (3.7). Note that accuracy is not well-defined in this setting, because the number of negative results (not detected arbitrary bounding boxes) is theoretically unlimited.

$$P = \frac{T}{D} \ , \quad R = \frac{T}{N} \ , \quad F_1 = \frac{2 * P * R}{P + R} \ . \tag{3.7}$$

Figure 3.7 illustrates two different ways of determining *true positive* detected subfigures for an example compound figure, which consists of three ground-truth subfigures: A, B, and C. We assume that a hypothetical CFS algorithm, given this compound figure as input, produced three subfigures – indicated as subfigures 1, 2, and 3 – at its output. In Figure 3.7, the resulting detected subfigures appear on the foreground, partially overlapping the three ground-truth subfigures in the background. The ImageCLEF and NLM evaluation protocols for this case will result in two different assessments, as follows:

- Figure 3.7 (a): The ImageCLEF evaluation procedure considers only one of subfigures 2 or 3 as true positive, depending on which of them gets associated first

with C. Note that both overlap ratios ρ^F_{C2} and ρ^F_{C3} are 100%. Subfigure 1, however, is regarded as false positive, because its overlap ratio, according to definition (3.6), with any ground-truth subfigure does not exceed 2/3. The resulting accuracy is therefore 1/3, since only one of the three detected subfigures qualifies as true positive.

- Figure 3.7 (b): The NLM evaluation procedure, on the other hand, determines that all detected subfigures should be considered false positives, because for subfigures 2 and 3 the overlap ratio, according to definition (3.5), with any ground-truth subfigure is too small (i.e., less than 75%), and subfigure 1 overlaps with two ground-truth subfigures (A and B) by at least 5%.

3.3.2.2 CFC-CFS Chain Evaluation

We propose to apply the CFS evaluation methods described above to the output of the CFC-CFS process chain (Section 3.2.3). Because CFS test datasets contain only compound figures, but the dataset for CFC-CFS chain evaluation also includes non-compound figures (Section 3.3.1), we need to extend CFS evaluation procedures by a convention to represent non-compound figures. We adopt the obvious solution to consider non-compound figures as "compound figures with a single subfigure" and represent each of them by a bounding box covering the entire image. This extension needs to be implemented in three different places of the evaluation procedure: (1) for ground-truth annotation, (2) for images classified as *non-compound* by CFC, and (3) for images classified as *compound* that are not divided into subfigures by CFS (because it does not detect proper separator lines).

Unmodified CFS evaluation algorithms can then be applied to the output of the CFC-CFS chain. Note that the ImageCLEF evaluation algorithm will assign 100% accuracy for true non-compound images only if there is exactly one "detected" subfigure in the CFC-CFS output, no matter what the bounding boxes are. Similarly, the NLM evaluation algorithm will find at most one true positive subfigure in a true non-compound image, but in this case the area of the "detected" bounding box is relevant (it must cover at least 75% of the entire image).

3.3.3 CFC-CFS Results

We conducted separate experiments to evaluate the proposed compound figure classifier (Section 3.3.3.1), our compound figure separation algorithm (Section 3.3.3.2), and the entire CFC-CFS process chain (Section 3.3.3.3). Our CFC and CFS approaches were compared separately to other existing methods, but evaluation of the CFC-CFS process chain has not yet been treated in the literature.

Table 3.4: Evaluation results of compound figure classifier on ImageCLEF CFC test set. From the 120 tested combinations of classifier algorithm, feature set, and number k of spatial bins, only the best three and the worst result for each classifier algorithm are reported. LogReg = logistic regression, SVM = support vector machine.

Classifier	*Feature Set*	k	*Accuracy%*	*FP%*	*FN%*
LogReg	134	16	76.9	16.9	6.2
LogReg	434	8	76.6	18.2	5.2
LogReg	434	16	76.6	17.7	5.7
LogReg	011	4	61.3	8.5	30.2
linear SVM	134	16	76.9	14.6	8.6
linear SVM	434	8	76.8	16.6	6.7
linear SVM	434	16	76.5	15.9	7.6
linear SVM	222	4	63.9	25.9	10.2
kernel SVM	034	4	75.5	20.4	4.1
kernel SVM	444	4	75.3	20.8	3.9
kernel SVM	434	4	74.2	23.0	2.9
kernel SVM	666	32	59.0	41.0	0.0

3.3.3.1 CFC Experiments

Results of CFC experiments are presented in Table 3.4. We used the ImageCLEF CFC dataset (Section 3.3.1) to train and evaluate the various combinations of feature sets and classifier algorithms described in Section 3.2.1. More specifically, we trained all three classifiers on 40 feature sets created by instantiating the 10 feature sets listed in Table 3.2 for four values of k (4, 8, 16, and 32). The quantization parameters were kept constant as $p = 5$, $q = 8$, and $h = 3$, as these values gave good classification performance in preliminary experiments. To enable a fair comparison with SVM, the logistic regression classifier used a probability threshold of 0.5, corresponding to a symmetric misclassification loss matrix (Eq. (3.2)) with $\alpha = 1$. From the 120 combinations of classifier algorithm, feature sets, and number k of spatial bins that were tested in experiments, we report only the best three and the worst results – separated by a dashed line in Table 3.4 – for each classifier algorithm with respect to accuracy.

Results indicate that feature set 434 achieves good classification performance for all three tested classifier algorithms with a rather low dimensionality of 96 (see Table 3.2). Feature set 134 (with 224 dimensions) with $k = 16$ spatial bins showed the same accuracy (76.9%) for both linear classifiers, becoming the best overall performer in both cases. The surprisingly low classification performance of kernel SVM is probably due to underfitting caused by default SVM hyperparameters; both box constraint C and

standard deviation σ of the radial basis function (RBF) kernel were kept at the default value 1.

Remarkably, the false positive rate of all well-performing classifiers in Table 3.4 is systematically higher than the false negative rate. This can be explained by two possible causes: first, the training set is slightly imbalanced (59% compound images), which may cause the classifier to decide in favor of the *compound* class in uncertain cases; second, the feature sets used for CFC produce a denser spatial distribution of non-compound images in the feature space than for compound ones, reinforcing the imbalanced training effect. In fact, the CFC features described in Section 3.2.1 have been designed to capture the existence of separators between subfigures. If such separators do not exist, feature values may exhibit a low variance across different images.

Compared to the best CFC run using visual-only features submitted to ImageCLEF 2015 by Wang et al. [222], which achieved 82.8% accuracy on the same dataset, our results are inferior by a margin of about 6%. However, as the approach of Wang et al. essentially employs a CFS algorithm (connected component analysis and band separator detection), we suppose that our CFC method has significant advantages with respect to efficiency for online classification. Extraction of the 111 feature set, which is the most complex of our proposed feature sets, took 81 milliseconds per image on average (excluding reading the image file from disk) using a MATLAB implementation on an Intel E8400 CPU operated at 3 GHz. This execution time corresponds to a processing rate of 12.3 images per second.

3.3.3.2 CFS Experiments

Experimental results of CFS evaluation using the ImageCLEF CFS dataset and corresponding evaluation procedure are shown in Table 3.5. The internal parameters of our CFS algorithm (see Appendix A.1), including implementation options of the illustration classifier (Section 3.2.2.1), were optimized on the training portion of the dataset, prior to evaluating CFS performance on the test dataset. For comparison, we also included a previous version of our approach [208] that did not use optimized parameters, and the best approach submitted to ImageCLEF 2015 (by NLM).

To analyze the effectiveness of the illustration classifier for CFS, we also report results for different classifier implementation options obtained by keeping these options constant during parameter optimization. Because logistic regression using *simple2* features was found to be most effective by parameter optimization when trained on the *greedy* set, we focused on this training set when evaluating other classifier implementations. Internal SVM parameters were optimized on the entire ImageCLEF 2015 multi-label classification test dataset (see Section 3.3.1) to maximize classification accuracy. The optimized `decision_threshold` parameter for deciding between edge-based

Table 3.5: Experimental results on the ImageCLEF 2015 CFS test set. Illustration classifiers are described in Section 3.2.2.1 (LogReg = logistic regression). BB denotes the percentage of images (or decisions*) where band-based separator detection was applied.

Algorithm	*Classifier*	*BB %*	*CFS Accuracy %*
Previous [208]	LogReg,simple2,first		49.4
NLM [178]	manual	95.7	84.6
Proposed	LogReg,simple2,first	61.6	84.2
Proposed	LogReg,simple2,majority	61.1	84.1
Proposed	LogReg,simple2,unanimous	61.8	84.2
Proposed	LogReg,simple2,greedy	75.8	84.8
Proposed	LogReg,simple11,greedy	74.1	**84.9**
Proposed	SVM,simple2,greedy	58.6	83.5
Proposed	SVM,simple11,greedy	60.3	83.5
Proposed	SVM,CEDD,greedy	59.2	82.8
Proposed	SVM,CEDD_simple11,greedy	59.6	83.2
Proposed	random,p=0.741	74.7	75.4
Proposed	no classifier,p=0	0	58.0
Proposed	no classifier,p=1	100	82.2
SubfigureClassifier	LogReg,simple11,greedy	60.1*	84.0

and band-based separator detection is effective only for logistic regression classifiers, because SVM predictions do not provide class probabilities.

Moreover, we consider a variant of the proposed CFS algorithm in which the illustration classifier has been replaced by a binary random decision unit, which predicts that a given input image is an *illustration* with probability p. For $p = 0$, the CFS algorithm will always use edge-based separator detection, and for $p = 1$ band-based separator detection will be applied to every input image. The value $p = 0.741$ corresponds to the decision rate of the most effective classifier (LogReg,simple11,greedy). The rationale for choosing p as the actual *illustration* decision rate of the classifier on the test dataset is to allow a fair comparison between the "random decision" variant and the proposed CFS algorithm, which should allow us to quantify the utility of the illustration classifier in our CFS approach.

The proposed CFS algorithm applies the illustration classifier once to each input image and reuses the classifier's decision in all recursive invocations of the separator detection module (see Fig. 3.3). To answer the question whether applying the classifier anew for each recursive invocation improves CFS performance, we also consider this algorithmic variant called *SubfigureClassifier* in our experiments, depicted in Fig. 3.8.

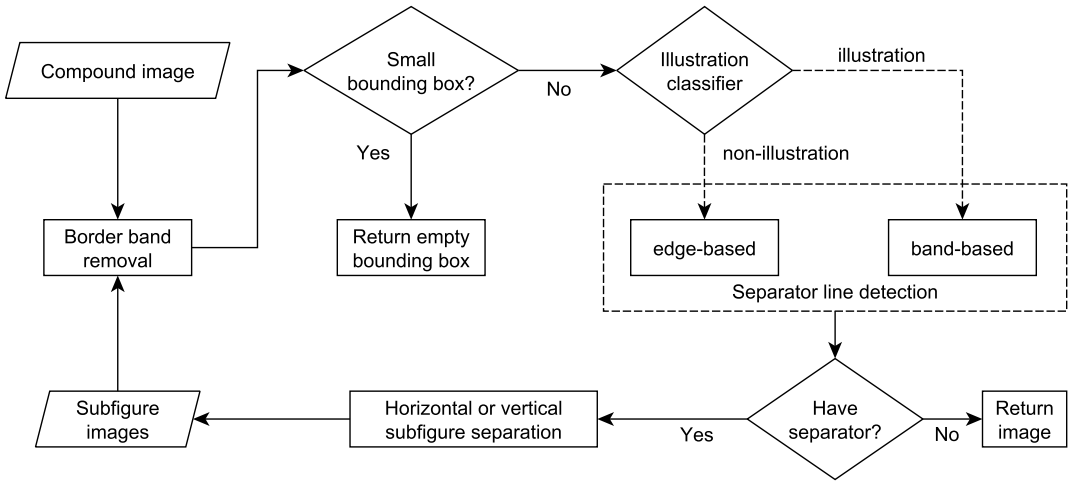


Figure 3.8: Variant of proposed CFS algorithm that applies the illustration classifier to every detected subfigure prior to splitting it further.

When comparing our results to NLM's approach, we note that the authors of [178] manually classified the test set into stitched (4.3%) and non-stitched (95.7%) images, whereas our approach uses automatic classification. Using band-based separator detection for all test images (no classifier, $p = 1$) works surprisingly well (82.2% accuracy), which can be explained by the low number of stitched compound images in the test set. On the other hand, using edge-based separator detection for all test images (no classifier, $p = 0$) results in modest performance (58% accuracy), which we attribute to a significant number of subfigures without rectangular borders (illustrations) in the test set. Selecting edge-based or band-based separator detection using the illustration classifier improved accuracy for all tested classifier implementations. In fact, it turned out to be effective to bias the illustration classifier towards band-based separator detection and apply edge-based separator detection only to high-confidence non-illustration images. This happened in two ways: by using the *greedy* training set, and by optimizing the `decision_threshold` parameter for the logistic regression classifier. This explains why best results were obtained by logistic regression classifiers trained on the *greedy* training set.

To further analyze the effectiveness of separator detection selection, we partitioned the CFS test dataset into two classes according to decisions of the most effective CFS algorithm variant (LogReg,simple11,greedy) and evaluated detection results of this algorithm separately on the two partitions. Resulting accuracy values of 85.7% on the edge-based partition and 84.6% on the band-based partition show that the classifier was successful in jointly optimizing detection performance for both separator detection algorithms.

Table 3.6: Evaluation results of compound figure separation on the NLM CFS dataset [7]. Precision (P), recall (R), and F_1 score are computed from the total number of ground-truth (G), detected (D), and true positive (T) subfigures.

Algorithm	*G*	*D*	*T*	*P%*	*R%*	F_1 *%*
Proposed (LogReg)	1656	1550	1314	84.8	**79.4**	**82.0**
Proposed (SVM)	1656	1584	1297	81.9	78.3	80.1
Apostolova et al. [7]	1764	1482	1276	**86.1**	72.3	78.6

Our algorithm was implemented in MATLAB and executed on a PC with 8 GB RAM and an Intel E8400 CPU running at 3 GHz. The average total processing time per compound image was 0.3 seconds when an illustration classifier with *simple* features was used, and 0.9 seconds when a classifier with CEDD features was applied. Note that the efficiency of other known approaches in the literature is either not documented [7] or by an order of magnitude lower ([44] reported 2.4 seconds per image).

To enable comparison with other CFS approaches in the literature, we further evaluated our approach on the NLM dataset using the evaluation procedure proposed by its authors (see Section 3.3.2). By using the same parameter values obtained by optimization on the ImageCLEF training set, CFS results on the NLM dataset provide additional information about the generalization ability of our CFS algorithm.

Results of evaluation on the NLM dataset are presented in Table 3.6. We selected the most effective illustration classifiers using logistic regression and SVM, respectively. They both use *simple11* features and the *greedy* training set. For convenience, we also included the results reported in [7] for a direct comparison with our approach.[3]

Results show that the relative performance of the proposed CFS algorithm using different classifiers is consistent with evaluation results on the ImageCLEF CFS dataset. The proposed algorithm could detect 10% more true positive subfigures than the image panel segmentation algorithm of Apostolova et al. [7], leading to a higher recall rate. On the other hand, precision is only slightly lower. Note that algorithm [7] has been used as a component in NLM's CFS approach [178] referenced in Table 3.5.

3.3.3.3 CFC-CFS Chain Experiments

To evaluate the effectiveness of the proposed CFC-CFS process chain, we used the CFC-CFS test dataset and evaluation procedure described in Sections 3.3.1 and 3.3.2,

[3]The dataset reported in [7] contains 400 images with 1764 ground-truth subfigures, so reported recall may be up to 0.4% higher if evaluated on the 398 images of the dataset available to us.

Table 3.7: Evaluation results of CFC-CFS chain for different algorithms and decision thresholds of the compound figure classifier (CFC). Decision thresholds are applicable to the logistic regression (LogReg) classifier only. The threshold marked by * was found to be optimal on the validation set. CR is the percentage of images classified as compound (compound figure rate). In addition to accuracy on the total test set, accuracy values on the subsets of predicted compound (C) and non-compound (NC) images are shown.

CFC	*Threshold*	*CR%*	*Accuracy%*		
			C	*NC*	*Total*
LogReg	0.20	84	84.7	94.7	86.4
LogReg	*0.35	74	84.9	90.8	86.5
LogReg	0.50	66	85.2	86.6	85.6
LogReg	0.65	56	85.9	81.1	83.8
linear SVM	–	61	85.6	82.1	84.2
kernel SVM	–	74	84.4	95.6	**87.3**
none	0	100	85.1	–	85.1
ideal		50	84.9	100	92.5

respectively. Results obtained using the ImageCLEF CFS evaluation method are presented in Table 3.7. For each of the three CFC algorithms (logistic regression, linear SVM, and kernel SVM) evaluated earlier, we applied the best-performing parameter settings according to Table 3.4. From these classifier algorithms, only logistic regression delivers predicted class probabilities, which allows to tune the effectiveness of the CFC-CFS chain by optimizing the decision threshold (Equation (3.4) in Section 3.2.3). Optimization was performed by evaluating CFC-CFS effectivness on the CFC-CFS validation set for decision thresholds d in the range $0.2 \leq d \leq 0.7$ using a step size of 0.05. The optimal value was found as $d = 0.35$, corresponding to weight $\alpha = 1.86$ of the misclassification loss matrix (Eq. (3.2)). In Table 3.7, we report results for four different decision thresholds on the test set. The optimal threshold selected during optimization on the validation set (indicated by *) also delivers best performance on the test set, confirming that improved performance for decision thresholds $d < 0.5$ is not caused by overfitting the validation set.

Column CR ("compound rate") of Table 3.7 shows the percentage of input images classified as *compound* by the different CFC implementations. Separate accuracy values on the portions of the test set classified as compound and non-compound, respectively, indicate a natural trend: accuracy increases with decreasing size of the class-specific subset. For logistic regression, the increase of accuracy on the non-compound subset for shrinking decision thresholds overcompensates the moderate loss on the compound subset, improving total accuracy. As the decision threshold approaches zero, however,

Table 3.8: Results of compound figure separation (CFC-CFS chain) on MCR dataset.

Processed article images	306,538	100%
Compound images predicted by CFC	222,042	72.4%
Compound images after CFS	154,136	50.3%
Total number of images after CFS	791,682	258%
Average number of subfigures per compound image	4.15	
Average number of images per article after CFS	10.6	

the number of predicted non-compound images and hence their effect on total accuracy becomes too small, leading to the observed local maximum of total accuracy for decision threshold $d = 0.35$.

High CFC-CFS accuracy on the subset of predicted non-compound images can also be explained by a low false negative rate of CFC: false negatives are true compound images classified as non-compound, which are not sent through CFS processing and hence hurt effectiveness of the CFC-CFS chain. This explains the good performance of kernel SVM in Table 3.7, although kernel SVM achieved inferior accuracy in CFC experiments (Table 3.4). From the three tested CFC algorithms, kernel SVM happened to have the lowest false negative rate at the cost of a high false positive rate, leading to a similar effect as decreasing the decision threshold for logistic regression.

From a wider perspective, however, effectiveness of CFC in the CFC-CFS process chain is rather limited when compared to processing all images of the test dataset with CFS only (indicated by classifier *none* in Table 3.7). In fact, our CFC implementations could improve CFC-CFS chain effectiveness by 2% only, whereas an *ideal* CFC algorithm that reproduces ground-truth class annotations would increase total accuracy by more than 7%.

Finally we note that all pairwise differences of total accuracy values in Table 3.7, which are mean values of accuracies determined for every input image, are statistically significant except for the difference between the first two lines in the table (logistic regression with decision thresholds 0.2 and 0.35, respectively). Significance has been tested at the 5% significance level using a paired t-test.

3.3.4 Compound Figures in MCR Dataset

To prepare further use of article images for medical case retrieval, we applied our CFC-CFS process chain to the MCR dataset, selecting the best performing CFC option according to Table 3.7, that is, kernel SVM with feature set 034 and $k = 4$ (cf. Table 3.4). Results are presented in Table 3.8. Compound figure separation enlarges the image set in this collection by a factor of 2.6, leading to more than 10 images per article on average.

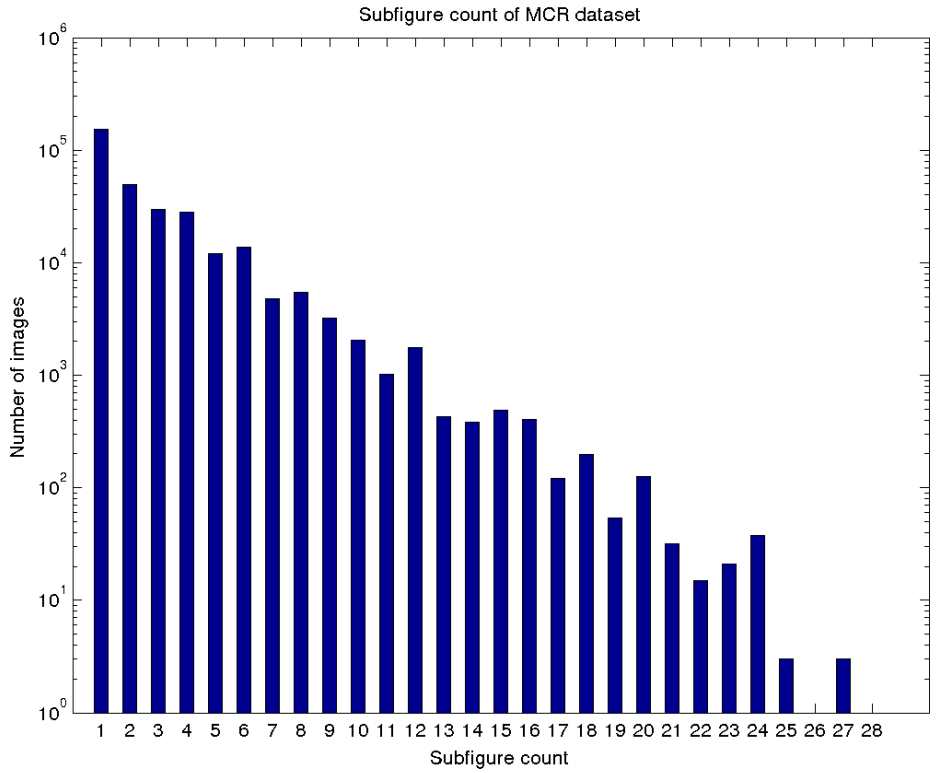


Figure 3.9: Distribution of subfigures per image recognized by our CFC-CFS process chain on the MCR dataset.

The compound figure rate of 50.3%, obtained by counting the number of images with more than one subfigure after applying the CFC-CFS chain, fits well into the range 40–60% reported in the literature for other datasets [7, 44, 88]. The distribution of recognized subfigures per image (including non-compound images) is depicted in Fig. 3.9. Note that the roughly linear representation in logarithmic scale corresponds to a Zipf distribution.

3.4 Summary

The high rate of compound figures (approximately 50%) found in scientific biomedical articles calls for an automated process to recognize and separate these figures prior to applying content-based analysis or indexing techniques. We proposed a two-step (CFC-CFS) process chain to automatically classify and separate compound images using light-weight image features and efficient image processing techniques. We evaluated our CFC and CFS approaches separately on public datasets and found that our fully automatic

CFS method delivered more accurate results than existing automatic or semi-automatic approaches. The inferior classification accuracy of our proposed CFC method, when compared to existing, more complex techniques, turned out to have only a limited effect on the effectiveness of the CFC-CFS process chain, which has not been investigated in literature so far.

We applied the proposed CFC-CFS process chain with best parameter settings found during evaluation to all article images of the MCR dataset in order to prepare further use of images to support medical case retrieval. We obtained a set of nearly 800,000 separated images, resulting in 10.6 images per scientific article on average.

CHAPTER

4 Biomedical Concepts

As every science, health care and life sciences have developed a large number of notions and terms used to describe their knowledge and subject of research. Starting with the need to index scientific literature to facilitate search and access in libraries, a plethora of controlled vocabularies and ontologies defining *biomedical concepts* have been developed during the last six decades (see Section 2.5). Nowadays, biomedical ontologies—or more generally, ontological artifacts—are used in a variety of applications that assist researchers or users in finding information and interpreting ever increasing amounts of biomedical data, including search in heterogeneous biomedical data, data exchange among applications, information integration, natural language processing, representation of encyclopedic knowledge, and computer reasoning with data [171].

This chapter investigates how biomedical concepts provided by a controlled vocabulary can be associated with medical case descriptions or case queries with the ultimate goal of enhancing the effectiveness of medical case retrieval (MCR). Since most of the documents of the MCR dataset (see Section 3.1) come annotated with MeSH (Medical Subject Headings) terms, we use the MeSH vocabulary for experiments, which is described in Section 4.1. We emphasize, however, that the proposed algorithms are not limited to MeSH by design and could easily be adapted to work with any other controlled vocabulary.

The main part of this chapter addresses the problem of automatic assignment of relevant biomedical concepts (MeSH terms) to given medical case descriptions or case queries, which we call *concept mapping*. Although the MCR dataset comes with MeSH annotations that have been generated by manual or semi-automatic procedures, we consider automatic concept mapping as a relevant problem for two reasons: (1) manual concept annotations may not be available for other datasets or other controlled vocabularies, and (2) manual concept annotations tend to be incomplete due to vocabulary size[1].

[1] Trieschnigg [212, p. 153] found that between 34% and 58% of MeSH terms that were predicted by automatic concept mapping, but did not correspond to manual annotations, were actually relevant to documents.

Given the multimodal representation of medical case descriptions (see Section 1.1), we consider three approaches to the concept mapping problem, differing in the modalities of information used as input: textual information (described in Section 4.2), visual information (Section 4.3), and both textual and visual information utilized by multi-view learning (Section 4.4).

Since we are interested in applying concept mapping as a means to enhance MCR, the effectiveness of concept mapping will be implicitly evaluated in experiments described in the following three chapters. In particular, the evaluation of concept-based retrieval (Chapter 6) may serve as the primary criterion to compare the effectiveness of different concept mapping algorithms for the purpose of MCR.

However, in this chapter (Section 4.5), we provide additional experimental results for text-to-concept mapping algorithms from a different evaluation perspective decoupled from retrieval. From a machine learning point of view, concept mapping may be regarded as a *multi-label classification* problem, where a given instance (medical case description) is to be assigned to one or more classes (biomedical concepts) whose labels are used to annotate the instance. Evaluation of classification performance requires the availability of ground-truth labels for a test dataset. For articles of the MCR dataset, we can use manual MeSH annotations as ground-truth labels for evaluation, which results in measuring the ability of a concept mapping algorithm to reproduce manual MeSH annotations.

Because manual MeSH annotations are not available for article images of the MCR dataset, this kind of evaluation is not possible for image-to-concept mapping algorithms. And since evaluation of the proposed multi-view concept mapping approach was left for future work (see Section 1.5), Section 4.6 summarizes experimental results for text-to-concept mapping approaches only, concluding this chapter.

Although the evaluation of some concept mapping algorithms is postponed to subsequent chapters or future work, the scientific contribution of this chapter is three-fold: (1) efficient novel text-to-concept mapping approaches based on string matching are proposed; (2) existing and proposed text-to-concept mapping systems are evaluated and compared on a biomedical dataset with respect to their ability to reproduce manual MeSH annotations; and (3) approaches to applying visual and multi-view concept mapping to a dataset of medical case descriptions are proposed.

4.1 Medical Subject Headings

Medical Subject Headings[2] (MeSH) are a controlled vocabulary (thesaurus) introduced in 1960 to index and catalog medical literature [48]. Since the first availability of MEDLINE, an online database of biomedical citations maintained by the U.S. National

[2] `http://www.nlm.nih.gov/mesh/`

Table 4.1: Number of terms contained in MeSH versions used for experiments.

MeSH version	*Primary terms*	*Synonyms*	*Synonyms per primary term*
2013	26,851	161,334	6.01
2014	27,149	191,839	7.07

Library of Medicine (NLM), in 1971, MeSH has been used to annotate its citations in a manual—and, many years later, semi-automatic—indexing process. For many years, searching MEDLINE by MeSH terms remained the primary search paradigm for biomedical literature. Even after fulltext search engines emerged in the 1990s, retrieval by MeSH terms often proved to be more effective for clinicians than fulltext search [37].

The MeSH thesaurus consists of *records* (see Fig. 4.1), each defining a single biomedical concept that may be referred to by one or more synonymous terms (called *MeSH terms*). One of the MeSH terms defined by a record is distinguished as the *primary MeSH term* (by the MH field, also known as *main heading*), the other MeSH terms are tagged as *synonyms* (by ENTRY fields). The MeSH vocabulary is updated by the NLM on a yearly basis to accommodate emerging and changing research topics in literature. Starting from 4,400 primary terms in 1960, the MeSH thesaurus has grown to include 27,883 primary terms in 2016. Experiments conducted for this thesis use either the 2013 or 2014 edition of MeSH, whose basic statistics are presented in Table 4.1.

Additionally, the MeSH thesaurus defines semantic relations between primary MeSH terms by assigning records to nodes of a graph that is constructed as a (non-disjoint) union of tree structures [151]. A parent node in a tree represents a more general term than its child nodes. The root nodes of all 16 tree structures of MeSH 2013 are listed in Table 4.2. Every MeSH record specifies one or more node identifiers (by *MN* fields) that define its position within the tree structures. The MeSH record shown in Figure 4.1 (primary MeSH term `Eye Neoplasms`), for example, defines two node identifiers C04.588.364 and C11.319, that assign it to successors of both nodes C04 (`Neoplasms`) and C11 (`Eye Diseases`) within the C tree structure (`Diseases`), as evident from Table 4.3. The number of dots in the node identifier is an indication of depth within the corresponding MeSH tree. We call it *MeSH node specialty*, as deeper nodes refer to more special MeSH terms, and present some examples in Table 4.3. The maximal MeSH node speciality occurring in the 2013 edition of MeSH is 11. As some proposed concept mapping algorithms need to assign a specialty value to a given MeSH term, which may have multiple node identifiers, we define *MeSH term specialty* as the average of specialty values of its MeSH nodes. So the MeSH term specialty of `Eye Neoplasms` is 1.5.

Today, most of MEDLINE publication records, and hence most articles of the MCR

```
*NEWRECORD
RECTYPE = D
MH = Eye Neoplasms
DE = EYE NEOPL
AQ = BL BS CF CH CI CL CN CO DH DI DT EC EH EM EN EP ET GE HI IM ME MI
MO NU PA PC PP PS PX RA RH RI RT SC SE SU TH UL UR US VE VI
PRINT ENTRY = Cancer of Eye|T191|NON|NRW|NLM (2000)|991103|abcdef
PRINT ENTRY = Eye Cancer|T191|NON|NRW|NLM (2000)|991103|abcdef
ENTRY = Cancer of the Eye|T191|NON|NRW|NLM (2000)|991103|abcdef
ENTRY = Neoplasms, Eye|T191|NON|EQV|NLM (2000)|991103|NEOPL EYE|abcdefv
ENTRY = Cancer, Eye
ENTRY = Cancers, Eye
ENTRY = Eye Cancers
ENTRY = Eye Neoplasm
ENTRY = Neoplasm, Eye
MN = C04.588.364
MN = C11.319
MH_TH = NLM (1966)
ST = T191
AN = coord IM with specific site in eye (IM) + histol type of neopl (IM)
MS = Tumors or cancer of the EYE.
...
UI = D005134
```

Figure 4.1: Partial MeSH 2013 record of `Eye Neoplasms`.

Table 4.2: Root nodes of MeSH 2013 tree structures.

A	Anatomy	I	Anthropology, Education, Sociology and Social Phenomena
B	Organisms	J	Technology, Industry, Agriculture
C	Diseases	K	Humanities
D	Chemicals and Drugs	L	Information Science
E	Analytical, Diagnostic, Therapeutic Techniques and Equipment	M	Named Groups
F	Psychiatry and Psychology	N	Health Care
G	Phenomena and Processes	V	Publication Characteristics
H	Disciplines and Occupations	Z	Geographicals

Table 4.3: Some primary MeSH terms and their positions in MeSH 2013 tree structures.

Primary MeSH Term	*Node Identifier*	*Specialty*
Eye Neoplasms	C04.588.364	2
Neoplasms by Site	C04.588	1
Neoplasms	C04	0
Eye Neoplasms	C11.319	1
Eye Diseases	C11	0
Kidney Pelvis	A05.810.453.537	3
Kidney	A05.810.453	2
Urinary Tract	A05.810	1
Urogenital System	A05	0

dataset (see Section 3.1), are annotated with MeSH terms, which can be retrieved using the Entrez search system API[3] [149]. We were able to retrieve MeSH annotations for 57,212 documents (76.6%) of the MCR dataset. They have been used as *manually annotated MeSH terms* in our experiments. Manual MeSH annotations come with an additional flag indicating whether a given MeSH concept represents a major topic of the document or not. We call these two types of annotations *major* and *minor* manual annotations, respectively.

4.2 Mapping Text to Concepts

Automatic prediction of MeSH terms that are relevant for a given biomedical publication or query has been an early research goal in the information retrieval field [10] (see also [194]). Existing concept mapping systems were designed to assist human annotators when assigning relevant MeSH terms to biomedical articles, or to help users of biomedical bibliographic databases to reformulate their queries using MeSH terms. Some of these systems, which were used in our experiments, are described in Section 4.2.1.

At the core of the most effective MeSH concept mapping systems is a class of simple machine learning algorithms known as *nearest neighbor classifiers*. They are based on the idea of instance-based learning, where the entire set of training instances is kept as the "learned model" and a new instance is classified by considering the classes of training instances that are "most similar" to the new instance. Section 4.2.2 explains

[3] `https://www.ncbi.nlm.nih.gov/books/NBK21081/`

how this approach can be applied to the multi-label classification problem of MeSH concept mapping.

All concept mapping approaches mentioned above are suitable for rather short documents or queries only, the application to longer documents would result in serious efficiency problems. To enable concept mapping for long documents applicable to a large collection of documents, we propose a novel family of efficient algorithms based on string matching that recognize MeSH terms (partially) occurring in documents. They are described in Section 4.2.3.

4.2.1 Existing Systems

An established concept mapping system developed by the U.S. National Library of Medicine (NLM) during the last two decades is *MetaMap*[4] [11]. It has been designed to map (short) biomedical text to concepts of the UMLS Metathesaurus in order to improve retrieval of MEDLINE citations, but it has been successfully applied also to other tasks utilizing biomedical concepts, like text mining, question answering, knowledge discovery, classification, and concept-based indexing of biomedical documents (most notably by NLM's Medical Text Indexer[5] web service).

The concept mapping approach of MetaMap relies on natural language processing techniques—including acronym and abbreviation identification, part-of-speech tagging, and shallow parsing—to identify phrases in the input text that contain words of the Metathesaurus. Linguistically inspired measures are then used to rank both UMLS concepts and combinations of concepts matching a given phrase. An optional word sense disambiguation (WSD) step favors concepts that are semantically consistent with surrounding text.

MetaMap is available as Java implementation for local installation and provides a number of configuration options to choose the vocabularies and data model to use, to specify the desired output, and to control algorithmic computations during concept mapping. For experiments, MetaMap was configured to restrict output to MeSH terms only and to display a ranked list of single concepts (as opposed to concept combinations). Other configuration options were left at their defaults. In particular, the WSD module was enabled.

According to the authors [11], MetaMap's strengths include the linguistically principled approach, its thoroughness when generating concept candidates, the evaluation metric used to rank concepts, its ability to find complementary combinations of relevant concepts, and its configurability that allows the tool to be applied to different tasks and domains. On the other hand, a limiting factor for many practical applications is time complexity, often prohibiting the use of MetaMap for processing long

[4] `https://metamap.nlm.nih.gov/`
[5] `https://ii.nlm.nih.gov/MTI/`

documents or a large collection of documents. While the authors report a processing time of "well under a minute" per citation in 2010 [11], we observed a mean execution time of 4.3 seconds per citation (consisting of title and abstract) in our experiments, resulting in an accumulated processing time of 89 hours for the entire ImageCLEF MCR dataset. Other weaknesses of MetaMap stated by its authors include its restriction to English text and a reduced accuracy in the presence of ambiguity, partly caused by the UMLS Metathesaurus.

Another existing concept mapping system used in a production environment is Open Biomedical Annotator (OBA) [103] developed by the National Center for Biomedical Ontology[6] (NCBO) at Stanford, USA. It has been designed as a web service to annotate textual datasets with concepts from a variety of biomedical thesauri and ontologies (provided by UMLS and the NCBO BioPortal[7]). OBA has been applied to annotate several dozens of public biomedical datasets (about 40 million records as of December, 2016) to construct a resource index that allows users of the NCBO BioPortal to search for biomedical records annotated with given concepts.

OBA employs a two-stage process for concept mapping: first, concept terms occurring in the input text (called *direct annotations*) are identified using a string matching approach implemented by *Mgrep* [184]; resulting concepts are then expanded with related concepts determined using hierarchical structures within ontologies or by mapping concepts between ontologies (as provided by the UMLS Metathesaurus). Details of the scoring function used by OBA to rank concepts in the result list are not clear [199].

OBA is available as a public RESTful (representational state transfer) web service[8] allowing to map single (short) text documents to biomedical concepts. Processing of long documents or batch processing of multiple documents is not supported by the public REST API. Parameters passed to an API call allow to select one or more ontologies (including MeSH) and the level for hierarchical concept expansion—level n selects all ancestors (more general concepts) at levels $l \leq n$ above direct annotations in the concept hierarchy. The resulting list of annotated concepts can be delivered in XML or JSON format. For experiments, only MeSH was selected as ontology and concept expansion was disabled (by setting $n = 0$).

The ability of OBA to produce relevant biomedical concepts has been evaluated and compared to MetaMap on both public biomedical datasets [184] and archives of a medical mailing list [199]. Both studies conclude that OBA's results yield higher precision values than MetaMap's, consistently across datasets and concept vocabularies. On the other hand, MetaMap often produced more relevant concepts than OBA in absolute numbers, indicating a higher recall, although recall could not be measured due

[6] `https://www.bioontology.org/`
[7] `http://bioportal.bioontology.org/`
[8] `http://bioportal.bioontology.org/annotator`

to missing expert judgments of false negatives. In addition, MetaMap's scoring function gives a better indication of concept relevance than OBA's [199].

Regarding efficiency, the string matching component (Mgrep) of OBA was reported to be an order of magnitude faster than MetaMap [184], and even OBA's web service was found to be more efficient than MetaMap [199]. However, we observed an average response time of about 12 seconds per citation (consisting of title and abstract) when using the REST API in our experiments, which was three times slower than MetaMap. Of course, the web service's response time depends on a number of external conditions like server load and network latency and hence is a bad indicator of OBA's efficiency, but it limits the applicability of this concept mapping system (via its public REST API) to mid-sized collections of short documents.

Another web service used in our concept mapping experiments is *Whatizit*[9], provided by the European Bioinformatics Institute[10]. The service allows to annotate (short) text with terms of various controlled biomedical vocabularies by selecting different text processing pipelines. For experiments, the *MeshUp* pipeline implementing a nearest-neighbor classifier (see Section 4.2.2) for MeSH concept mapping [211] was used, but this specific pipeline was taken offline soon after most of our experiments were done in 2015.

4.2.2 Nearest-Neighbor Classifiers

Nearest-neighbor classifiers [14, 84] for text-to-concept mapping utilize a collection of documents already annotated with concepts by some other means as training set, which in combination with a similarity measure on text documents serves as a classifier model. To map a given input text to MeSH concepts, the "most similar" documents in the collection are retrieved and their annotated MeSH concepts are used to classify the input text. Retrieving similar documents is achieved effectively and efficiently by a classical information retrieval system. A simple and common strategy to decide which retrieved documents are "most similar" is to apply a rank threshold k to the ranked list of retrieved documents. The resulting classifier is therefore also called *k-nearest neighbor* (kNN) classifier.

Another strategy is needed to select MeSH concepts from the k nearest neighbor documents, in order to classify the input text. For experiments, we ranked all MeSH concepts appearing as annotations of k nearest neighbor documents—we call them *candidate MeSH concepts*—, and applied a rank threshold m to the ranked list of candidate concepts. Ranking of candidate concepts is achieved by accumulating the retrieval scores of containing documents and taking into account the reliability of concept annotations.

[9] `http://www.ebi.ac.uk/webservices/whatizit/info.jsf`
[10] `http://www.ebi.ac.uk/`

Experiments used the following reliability factors (given with their default values) for different types of MeSH annotations:

- *Major manual MeSH annotations:* $\alpha_1 = 1.5$. Annotations contained in MEDLINE publication records flagged as *major topic* (see Section 4.1) receive the highest reliability factor.
- *Minor manual MeSH annotations:* $\alpha_2 = 1.2$. Non-major manual annotations are assigned a smaller reliability factor.
- *Automatic MeSH annotations:* $\alpha_3 = 1.0$. Annotations created automatically by a concept mapping algorithm receive the lowest reliability factor. For efficiency reasons, only string matching approaches (see Section 4.2.3) were applied for automatic document annotation.

To define how the score of a candidate concept c is calculated, suppose that c appears in MeSH annotations of a set R_c of retrieved documents within the top k ones ($|R_c| \leq k$), let s_d be the retrieval score of document d, let $t(c, d)$ be the MeSH annotation type of concept c in document d and $\alpha_{t(c,d)}$ its reliability factor. The score S_c of concept c used to rank concepts is then calculated as:

$$S_c = \sum_{d \in R_c} \alpha_{t(c,d)} \, s_d \tag{4.1}$$

The described ranking strategy for candidate MeSH concepts assigns higher scores to concepts that appear in annotations of multiple documents, and prefers manual annotations over automatic annotations. Optimal values for threshold parameters k and m depend on the document collection and need to be determined using a validation set.

Nearest neighbor classifiers are the most effective text-to-MeSH concept mapping algorithms known so far [70, 212] when measuring their ability to reproduce manual MeSH annotations. Effectiveness is limited, however, when the input text represents a topic that is not covered by the document collection—which can be fixed by choosing a larger or more suitable document collection. From an efficiency point of view, kNN classifiers often provide higher processing rates than approaches based on natural language processing (see Section 4.2.1), because text retrieval systems were designed to retrieve k-nearest neighbor documents efficiently. On the other hand, IR systems were optimized for short queries and may be slow or unusable for long input documents, leading to a similar limitation as recognized for existing concept mapping systems described in Section 4.2.1. Currently, only text-to-concept mapping approaches based on string matching seem to overcome the obstacle of long input documents. We therefore describe our string matching approach to MeSH concept mapping in the next section.

4.2.3 String Matching

String matching approaches to concept mapping try to detect literal occurrences of concept names in a given text document. Since biomedical concepts, especially MeSH terms, often consist of multiple words and not all possible lexical variations of concept names (due to flexion, word order, whitespace or punctuation variations) may be contained in the concept thesaurus, traditional string or pattern matching algorithms are likely to be ineffective. Moreover, partial matches—e.g. two of three words constituting a biomedical concept—would be missed. And finally, traditional pattern matching algorithms would simply be too inefficient to search for occurrences of more than 160,000 different MeSH terms in a document or in a large collection of documents. For these reasons, indexing techniques used in information retrieval systems provide a more suitable approach to concept mapping based on string matching.

We could use an existing IR system to index MeSH terms and present the document to be mapped to concepts as a query to the system, which would retrieve a ranked list of MeSH terms. However, this method is likely to be ineffective and inefficient, because retrieval systems have not been designed to index very short "documents" (i.e. MeSH terms) and to execute queries that may well be longer than the average length of indexed "documents".

We therefore developed several MeSH term matching algorithms that enable an efficient generation of a ranked list of MeSH terms supposed to be relevant for a given (long) document. All algorithms work by accumulating MeSH term scores during a single pass through the document, followed by score normalization and optional MeSH term specialty boosting. The latter method favors MeSH terms at greater depth in a MeSH tree, i.e. more special MeSH terms are preferred over more general terms containing the same words. The algorithms are listed below. Their components are described in the following sections.

t0 – BinCov binary coverage

t1 – Dist distance-based match frequency

t2 – BinDist combination of *BinCov* and *Dist* for matching runs

t3 – IdfBinDist *BinDist* with score boosting by maximal IDF of MeSH term words

t4 – IdfCovDist combination of *Dist* with IDF-based run coverage

4.2.3.1 Basic Algorithm and Data Structures

For the purpose of MeSH term matching, the notion of a *MeSH term* always refers to a single synonym of a MeSH record (see Section 4.1), that is, MeSH term matching is performed on lexical entities, not on semantic concepts.

All algorithms use an inverted index of MeSH term words. Every word of the MeSH thesaurus is linked to a list of MeSH terms containing that word. When building the index, words are lower-case-filtered, and punctuation characters are removed. Stop words are not removed, because they may be significant for a MeSH term (as in `Vitamin A`). Since MeSH often contains singular and plural forms of MeSH terms as synonyms, and to favor exact matches, word stemming is not applied.

When processing a document, the same preprocessing is applied as for building the inverted index, and for each word of the document all MeSH terms containing that word are visited. Visited MeSH terms maintain local statistics depending on the algorithm in use. When document processing has finished, all visited MeSH terms are updated to produce final scores by performing score normalization and specialty boosting. Finally, visited MeSH terms are sorted by score, and the list of matching MeSH terms is obtained by thresholding the score. In fact, the implementation uses a priority queue to assemble the final sorted list of MeSH terms to avoid sorting all visited MeSH terms.

MeSH term matching algorithms differ only in the way they accumulate statistics and compute the final score of visited MeSH terms. The different scoring functions are described in the following sections.

4.2.3.2 Coverage

We define the ratio of words of MeSH term t occurring in a document d as the *coverage* $\mathbf{Cov}(t, d)$ of this MeSH term in the document. Word order and number of occurrences of the same word are ignored. For example, given the document "Abdominal CT scan revealed a large left renal mass with extension into the left renal pelvis and ureter.", the coverage of MeSH term `Pelvis, Renal` is 1.0 and that of MeSH term `Pelvis Cancers` is 0.5. This scoring function makes sense only for very short documents, as longer documents will raise the scores of many irrelevant MeSH terms to 1.0, because their constituent words are spread over the entire document.

An even simpler scoring function that is only used in combination with other functions described below is the *binary coverage* $\mathbf{BinCov}(t, d)$. It is defined as 1 when all words of MeSH term t occur in document d, and 0 otherwise.

4.2.3.3 Distance-Based Match Frequency

To make MeSH term matching sensitive to word order and to the proximity of MeSH term words occurring in the document, we define the score as a function of relative positions of MeSH term words in the document. Let $t = t_1\, t_2\, \ldots\, t_T$ be the constituent words of MeSH term t, $p_1 < p_2 < \cdots < p_N$ the word positions within document d containing MeSH term words t_i, and $r_1, r_2, \ldots, r_N$ the corresponding MeSH term word indexes, i.e. the word at document position p_i is MeSH term word t_{r_i}. (If the MeSH

term t contains the same word at multiple positions and this word occurs at position p_i in the document, then we define r_i as the minimum of those positions in t.) The scoring function $\mathbf{Dist}(t, d)$ is then defined as follows:

$$s(p, r) = \begin{cases} (p\, r)^{-1} & \text{if } r > 0, \\ 0 & \text{if } r = 0, \\ (p\, (2 - r))^{-1} & \text{if } r < 0. \end{cases} \tag{4.2}$$

$$\mathbf{Dist}(t, d) = \begin{cases} \sum_{i=1}^{N-1} s(p_{i+1} - p_i,\, r_{i+1} - r_i) & \text{if } N > 1 \text{ and } T > 1, \\ 0 & \text{if } N = 1 \text{ and } T > 1, \\ N & \text{if } T = 1. \end{cases} \tag{4.3}$$

Note that $s(p, r)$ is defined for $p > 0$ only, and $s(p, r) > s(p, -r)$ if $r > 0$. The intention behind these formulas is that M exact occurrences of the MeSH term in the document shall give a score of approximately $M\,(T - 1)$ if $T > 1$, but shall allow also for partial matches and word re-orderings with a penalty. The scoring function can therefore be viewed as a distance-based *soft match frequency* of MeSH term words. The score is not normalized with respect to MeSH term length T in order to favor longer MeSH terms.

For example, when calculating the score of MeSH term `Pelvis, Renal` for the short document of the previous section, we have $p_1 = 8$, $p_2 = 15$, $p_3 = 16$ and $r_1 = 2$, $r_2 = 2$, $r_3 = 1$, resulting in the score $0 + 1/3 = 0.333$. MeSH term `Pelvis Cancers` has score 0 for the same document, because `pelvis` occurs only once and `cancers` does not occur.

4.2.3.4 Run Coverage and Match Frequency

The **Dist** scoring function described in the previous section may give rather high values for MeSH terms containing some frequently occurring word groups, although the entire MeSH term is not contained in the document. The most prominent such word group is `of the`, which is part of many MeSH terms (e.g. `Cancer of the Uterus`, `Infarct of the Spleen`, `Exstrophy of the Bladder`). To address this problem, we introduce the notion of *matching runs* and restrict the **BinCov** and **Dist** scoring functions to those runs.

Using the notation of the previous section, we define a *matching run* as a maximal subsequence $(p_i,\, p_{i+1},\, \ldots,\, p_k)$ of matching positions of a MeSH term in a document, such that $p_{j+1} - p_j \leq \beta$ for all $j \in [i, k - 1]$ and a fixed parameter β. For experiments, a default parameter value of $\beta = 3$ was used. Matching runs are groups of consecutive matching positions separated from other such groups by more than β positions. Note that the boundaries between matching runs can be easily determined during a single pass through the document.

The **BinDist** scoring function is computed from products of **BinCov** and **Dist** functions restricted to matching runs $\pi_1, \ldots, \pi_R$ of MeSH term t in document d:

$$\mathbf{BinDist}(t,d) = \sum_{i=1}^{R} \mathbf{BinCov}(t,\pi_i)\ \mathbf{Dist}(t,\pi_i) \tag{4.4}$$

The restriction of binary coverage to matching runs is called *run coverage*. If run coverage is 1 for all matching runs, the **BinDist** score will approximate the **Dist** score, because the run distance β will limit inter-run contributions of $\mathbf{Dist}(t,d)$ to small values. The **BinDist** scoring function effectively ignores all partial occurrences of a MeSH term in the document, but allows for word permutations and intermixing with other words within matching runs.

For example, considering the short document d given in Section 4.2.3.2 and MeSH term $t =$ `Pelvis, Renal`, there are two matching runs for $\beta = 3$: $\pi_1 = (8)$, $\pi_2 = (15, 16)$. We have $\mathbf{Dist}(t,\pi_1) = 0$, $\mathbf{Dist}(t,\pi_2) = 1/3$, and $\mathbf{BinCov}(t,\pi_2) = 1$, so $\mathbf{BinDist}(t,d) = 0.333$.

4.2.3.5 Boosting MeSH Terms by IDF

A major problem with scoring functions based on match frequency is that one-word MeSH terms occurring several times in a document obtain higher scores than multi-word MeSH terms occurring only once. However, the long MeSH term may be equally relevant, because it denotes a medical concept that is rarely mentioned in the document collection. On the other hand, many one-word MeSH terms occur in a large fraction of documents in the collection, so their importance of being relevant for a given document should be decreased. This observation calls for integration of *inverse document frequency* (IDF) of MeSH terms into the scoring function, which takes greater values for MeSH terms occurring less frequently in the document collection.

When defining IDF of MeSH terms, we need to take into account that not all MeSH terms occur in the document collection at hand, and that counting the document frequency of MeSH terms may require automatic MeSH term matching, resulting in a recursive problem. Additionally, the question of how to count synonyms of MeSH terms should be answered. We worked around these problems by defining the *IDF of a MeSH term* as the maximal IDF value of its constituent words. That is, we reduce the global importance of a MeSH term to its most discriminative word with respect to the collection.

The IDF value of a MeSH term word remains to be defined as it may not occur in the document collection at all. Additionally, we have to take care of stop words (e.g. `of` and `the`), which are usually not indexed or counted by the retrieval system used to index the document collection. Let w denote a word of a MeSH term, let N be the number of documents in the collection, and n_w the document frequency of w in the collection (i.e.

the number of documents containing w) if w has been indexed by the retrieval system. We call w a *collection stop word* if it is a common English stop word or if it occurs in all N documents of the collection. If w does not occur in the collection (and hence is not a common English stop word with high probability), we call it an *external term*.

$$\mathrm{IDF}(w) = \begin{cases} \varepsilon & \text{if } w \text{ is a collection stop word,} \\ (\log N)/2 & \text{else if } w \text{ is an external term,} \\ \log(N/n_w) & \text{otherwise.} \end{cases} \tag{4.5}$$

We assign some small positive IDF value $\varepsilon < 1$ (we used $\varepsilon = 0.1$ in our implementation) to collection stop words, for reasons explained in the next section. External terms receive half of the maximal IDF value possible for collection terms. Note that $\mathrm{IDF}(w) > 0$ in all cases. The IDF value of MeSH term $t = t_1\, t_2\, \ldots\, t_T$ is defined as explained earlier and used to boost the **BinDist** score:

$$\mathrm{IDF}(t) = \max_i \mathrm{IDF}(t_i) \tag{4.6}$$

$$\mathbf{IdfBinDist}(t, d) = \mathrm{IDF}(t) \cdot \mathbf{BinDist}(t, d) \tag{4.7}$$

4.2.3.6 IDF-Weighted Run Coverage

The binary run coverage used by **BinDist** and **IdfBinDist** scoring functions effectively ignore partial matches of MeSH terms in a document, in the sense that runs missing only one word of a MeSH term do not contribute to the matching score. However, such runs can be regarded as relevant for the MeSH term if the missing word has low discriminative power in the document collection, which is the case for e.g. collection stop words (see Section 4.2.3.5).

An alternative approach to improving the **BinDist** scoring function is to allow this kind of partial matches to contribute to the score. This is achieved by replacing the binary run coverage by an *IDF-weighted run coverage* **IdfCov** of matching runs $\pi_1, \ldots, \pi_R$ of MeSH term $t = t_1\, t_2\, \ldots\, t_T$ in document d:

$$\mathbf{IdfCov}(t, \pi) = \frac{\sum_{i=1}^{T} \mathrm{IDF}(t_i) \cdot \mathbf{BinCov}(t_i, \pi)}{\sum_{i=1}^{T} \mathrm{IDF}(t_i)} \tag{4.8}$$

$$\mathbf{IdfCovDist}(t, d) = \sum_{i=1}^{R} \mathbf{IdfCov}(t, \pi_i) \cdot \mathbf{Dist}(t, \pi_i) \tag{4.9}$$

The binary coverage $\mathbf{BinCov}(t_i, \pi)$ is 1 if MeSH term word t_i occurs in matching run π, and 0 otherwise. $\mathrm{IDF}(t_i)$ has been defined in Equation (4.5), and the **Dist** scoring function is the same as in Section 4.2.3.4. The definition of **IdfCov** also explains why

$\mathrm{IDF}(t_i)$ has been defined to be positive for all MeSH term words t_i: in addition to providing mathematical validity of the fractional expression, it guarantees a penalty for missing MeSH term words in matching runs.

4.2.3.7 Boosting MeSH Term Specialty

It is reasonable to assume that more special MeSH terms are more relevant to a document, even if they occur less often in the document than more general MeSH terms. We therefore equipped all MeSH term scoring functions described in the previous sections with an optional boost factor based on MeSH term specialty as defined in Section 4.1. So for any scoring function $\mathbf{score}(t, d)$ defined above we also consider a variant $\mathbf{score}_s(t, d)$ boosted by MeSH term specialty $\mathrm{spec}(t)$:

$$\mathbf{score}_s(t, d) = \alpha^{\mathrm{spec}(t)} \cdot \mathbf{score}(t, d) \tag{4.10}$$

where $\alpha > 1$ is a fixed parameter (we used $\alpha = 1.3$ in our experiments).

4.3 Mapping Images to Concepts

The large number of concepts supposed to be supported by image-to-concept mapping approaches prohibits the direct application of known visual classifier algorithms that learn a classifier model from content-based image features (see Section 2.3). A feasible approach, however, is provided by nearest neighbor classifiers that retrieve annotated images that are "similar" or semantically related to the given input image and use their concept annotations to label the input image.

Candidate concepts can then be ranked and selected as described in Section 4.2.2. However, since retrieval scores of images tend to be less reliable than retrieval scores of text documents, we calculated the score of retrieved images by a reciprocal rank function:

$$t_d = \frac{1}{9 + r_d} \tag{4.11}$$

where t_d is the score of a retrieved image d (taking the role of retrieved documents in Equation (4.1)), and r_d is the rank of d. The top-most image has rank 1, which translates into a maximal score of 0.1. The additive constant 9 reduces score differences between images near the top of the ranked list.

In our experiments, we generated concept annotations of images by applying text-to-concept mapping based on string matching (see Section 4.2.3) to image captions, which are available for all images of the MCR dataset. MeSH annotations were then

stored in an image index created using the Lucene Image Retrieval Library[11] (LIRE), together with searchable image descriptors. We consider three types of image descriptors, corresponding to different features and similarity measures used for image retrieval:

- *Global image features:* These correspond to well-known image descriptors representing the entire image and readily available with LIRE. In experiments we used CEDD [40] and FCTH [41] only, but other global features could be used as well. The Euclidean distance in feature space is used as a dissimilarity measure for retrieval.

- *Global feature mapping:* This approach combines multiple global image features with unsupervised learning (clustering) to represent an image by a combination of synthetic textual identifiers (called *visual code words*) corresponding to cluster centers [188]. Image retrieval can therefore use efficient text retrieval methods, and the additional layer of abstraction introduced by clustering may help to bridge the semantic gap. However, due to time constraints, we were not able to implement this approach for experiments.

- *Concept vectors:* If the input image can be associated with meaningful concepts by some other means (e.g. using the textual information of a case query), then concept-based retrieval (see Chapter 6) can be applied to retrieve semantically related images for nearest-neighbor classification. Optionally, retrieved images can be reranked by visual similarity with the input image using a content-based descriptor. In experiments, we associated the images of a case query with MeSH concepts obtained by processing the query text with a kNN classifier, and used global image features (CEDD, FCTH, PHOG [27]) for reranking.

Due to the lack of ground-truth annotations for images of the MCR dataset, classification performance of image-to-concept mapping algorithms could not be evaluated directly, but their effectiveness for concept-based retrieval will be investigated in Chapter 6.

4.4 Multi-View Concept Mapping

When the item to be mapped to concepts is represented by multimodal data (e.g. text and images of a case query), the correlation and complementary information of different modalities may help to implement a more effective multi-label classifier for concept mapping. This is the objective of multi-view learning approaches (see Section 2.6), which take a slightly more general perspective by considering multiple representations

[11] `http://www.lire-project.net/`

(*views*) of the same instance for joint learning. In this section, we propose a concept mapping approach based on multi-view learning and address some practical problems that arise when the approach should be applied and evaluated on the MCR dataset. Although we were not able to implement the proposed approach due to time constraints, we include the description as a basis for future work.

We propose to apply the Coupled Dictionary Learning and Feature Mapping method of Xu et al. [Xu2015] for multi-view concept mapping, because (1) it learns low-dimensional correlated sparse representations of views (eliminating the need for pre-processing raw features), (2) it learns a low-rank mapping of sparse representations into a space of semantic labels (which can be MeSH concepts for our purposes), and (3) it represents a recent successful approach of multi-view subspace learning that has been evaluated on a medium-sized dataset of images (MIRFlickr-25K [95]).

However, as for many other existing multi-label classification approaches, the total number of different labels (classes) in the datasets used for evaluation is small (< 40), whereas the MeSH thesaurus contains approximately 27k concepts, leading to a severe *training sample size* problem. That is, the number of available training samples for a given concept may be too small to obtain a robust classification model. We therefore need to develop a strategy for appropriately reducing the number of MeSH terms used for concept mapping.

Another problem that needs to be addressed is the *granularity mismatch* between textual and visual representations of medical case descriptions in the MCR dataset (biomedical articles or case queries): such descriptions consist of text and zero or more images, while multi-view learning algorithms assume that every instance is represented by a fixed number of views, even if they tolerate missing view representations for some instances.

The training sample size and granularity mismatch problems can be addressed by appropriate dataset preprocessing methods, as described in Section 4.4.1. Section 4.4.2 gives some hints for implementing the chosen multi-view learning approach, and the actual concept mapping procedure is described in Section 4.4.3.

4.4.1 Dataset Preprocessing

The proposed preprocessing of the MCR dataset for applying and evaluating the multi-view concept mapping approach is depicted in Fig. 4.2. Compound figures in article images are recognized and separated as described in Section 3.2. Prior to building view representations that will be used for multi-view learning, textual and visual content need to be represented by compact descriptors (one feature vector per document or image) produced by *feature extraction* modules. Although proper feature selection would require separate evaluation, we limit ourselves to a few feature representations that we consider effective for the purpose of multi-view concept mapping:

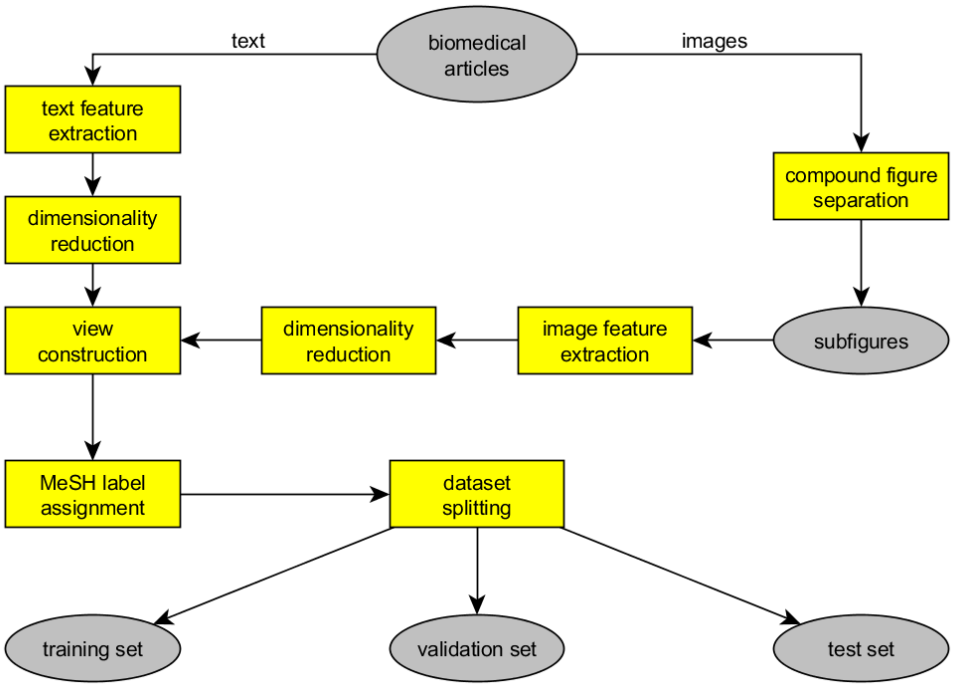


Figure 4.2: Dataset preprocessing for multi-view concept mapping.

- **Textual features:** Term vectors according to the TF-IDF vector space model of Lucene can be used (see Section 2.2.1). English words are stemmed, stop words and other high-frequency (w.r.t. document frequency) words are removed from the vocabulary to limit the dimensionality of the feature space, which depends on the training document collection. We note that the vector space model is still a competitive IR model on general text corpora compared to other established IR models (probabilistic models, language models, divergence from randomness) [13, Sect. 3.2.8][174][242].

- **Visual features:** We propose to use a concatenation of the following image descriptors, which capture global image characteristics of color, texture and shape as well as local characteristics extracted from salient image patches: (number in parentheses denote the dimensionality of the descriptor)

  - *CEDD* (144): color and edge directivity descriptor [40], available in LIRE.
  - *FCTH* (192): fuzzy color and texture histogram [41], available in LIRE.
  - *BTDH* (768): brightness and texture directivity histogram with Z-grid fractal scanning [39]; was shown to be effective for medical image retrieval; available as C# code only.

- *SIFT-VLAD* (64*128=8192): local SIFT features aggregated using the VLAD model [99] and a codebook of 64 visual words (produced by k-means clustering); was shown to be more effective for image retrieval than the popular bag-of-visual-words aggregation of SIFT features with much larger codebook sizes [99]; available in LIRE.
- *SIMPLE-CEDD-VLAD* (64*144=9216): CEDD features extracted from random local patches[12] [98], aggregated using VLAD as for SIFT-VLAD; available in LIRE.
- *Thumbnail32* (1024): image resized to 32×32 pixels; proved to be effective for image categorization on IRMA dataset of medical images [60].

After text feature extraction from the MCR dataset we obtained term vectors of large dimensionality (900k), which presumably cannot be used directly as text feature vectors for multi-view learning. Xu et al. [235] used rather low-dimensional (< 500) text features in their experiments, but on the other hand, worked with visual features of dimensionality 7500. We therefore aim to reduce the dimensionality of textual and visual features to values below 10k prior to multi-view learning. We propose slightly different methods for *dimensionality reduction* (DR) of textual and visual features:

- *DR of textual features:* Following the method applied in a different multi-view learning approach [75], dimensionality reduction is achieved in two steps: (1) select the K most frequent terms (e.g. K =30k) with respect to document frequency, after removing terms that occur in nearly all documents of the corpus (say, in more than 80% of documents); (2) apply sparse SVD (equivalent to PCA, but more efficient and available in Matlab) to further reduce dimensionality to, say, 3000.
- *DR of visual features:* If all proposed visual descriptors were concatenated, we would obtain 19536-dimensional feature vectors. We therefore suggest to reduce the dimensionalities of the two VLAD descriptors using PCA to 500 each (again following [75]). The combination of all descriptors then results in a dimensionality of 3128.

The *view construction* module assembles textual and visual feature vectors originating from the same source instance (medical case description) into a fixed number of view representations, thereby addressing the granularity mismatch problem mentioned in the beginning of Section 4.4. We propose several alternatives for view construction:

[12] according to LIRE documentation, random local patches delivered better results on general image datasets than using the SURF keypoint detector.

- *Text replication:* Every image of a case description (or case query) D is associated with 2 views: visual features, and the entire text of D. That is, a case description with k images results in k replicated cases. This strategy will increase the dataset size by a factor $\bar{k}$, where $\bar{k}$ is the average number of images per original case. The same is true for the query set.

- *Using image captions:* Every image of a case description D is represented with 3 views: visual features, image caption text, title and abstract of D. This leads to the same increase in dataset size as for text replication. Case queries are represented by two views only (textual and visual features), which does not present a problem, because they are not used for training and the selected multi-view learning approach [235] can cope with missing view representations.

- *Image selection:* Cluster all images of the dataset (or a large random sample) into a small number K (say $K = 2$ or $K = 3$) of clusters using global visual features, and associate each case description D with at most $K+1$ views: the entire document text, and one image (visual features) contained in D selected randomly from each cluster. This may result in some case descriptions (and queries) having less than $K + 1$ view representations; use only those with exactly $K + 1$ views for training. The rationale behind this strategy is that clusters are expected to represent different image modalities (e.g. medical images and illustrations), resulting in more coherent and discriminative representations of a single view. This strategy does not increase the dataset or query set size.

We call an entity represented by multiple views (e.g. document text and one image) a *data sample*, so the view construction process results in a set of data samples. For training, parameter selection, and evaluation, data samples need to be associated with ground-truth labels, i.e. with relevant MeSH terms. For the majority of articles of the MCR dataset manual MeSH annotations are available (see Section 4.1), but for application to concept-based retrieval (Chapter 6) it may be beneficial to automatically add other relevant MeSH terms to data samples for two reasons: manual annotation often misses relevant MeSH terms due to MeSH vocabulary size (as noted in the introduction to this chapter on page 61), and depending on the chosen multi-view representation, a data sample may refer to a certain article image whose caption may give rise to additional MeSH terms not covered by document-level MeSH terms. We therefore propose to evaluate three strategies for *assigning MeSH labels to multi-view representations*:

1. Just use the manually annotated document-level MeSH terms, and restrict the training set to instances having such annotations.

2. In addition to manually annotated MeSH terms, use MeSH concepts obtained from automatic text-to-concept mapping of image captions (see Section 4.2).

3. In addition to manually annotated MeSH terms, use MeSH concepts obtained from automatic text-to-concept mapping of the entire case description (including image captions).

Additionally, MeSH label assignment can help to resolve the *training sample size* problem mentioned in the beginning of Section 4.4. We propose to exploit the hierarchical structure (directed acyclic graph, DAG[13]) of the MeSH thesaurus to reduce the number of MeSH concepts included in the classifier model in a data-driven manner. Algorithm 1 presents the proposed MeSH label reassignment algorithm, which ensures that there are at least T data samples per selected MeSH concept.

The algorithm pushes data samples up in the MeSH DAG until at least T samples are assigned to a single node m. Note, however, that data samples assigned to m are not pushed further up towards a root node, but the process of collecting data samples starts anew when ascending from m to a root node. We think that partitioning the training set in this manner among MeSH nodes is more effective for learning a multi-label classifier than duplicating data samples in more general MeSH nodes (closer to a root node). Also note that a small number of data samples may end up in root nodes m with $|B(m)|| < T$ and hence will be ignored for training, but as root nodes correspond to very general MeSH concepts, loosing them should not present a problem.

The final step during dataset preprocessing is *dataset splitting*, which randomly partitions the set of annotated data samples resulting from MeSH label assignment into training, validation, and test subsets. Given the MCR dataset with approximately 57k manually annotated articles (resulting in a still larger number of data samples, depending on the view construction process), we propose to randomly select 5000 data samples for validation, another 5000 for testing, and the remainder for training the multi-view classifier.

4.4.2 Multi-View Learning Implementation

The chosen multi-view learning algorithm [235] proceeds in two phases: (1) sparse representations of each view are generated by coupled dictionary learning, and (2) low-rank projections from sparse representations to (MeSH) concept space are learned by coupled feature selection. Although Xu et al. [235] do not provide the code of their approach, separate Matlab code for subtasks (1) and (2) is provided by authors of earlier papers [94, 220] that Xu et al. rely on. A brief inspection showed that both pieces of Matlab code can be integrated to implement the complete approach of [235]. The Matlab code depends on the open-source Sparse Modeling Software toolbox[14] (SPAMS).

[13] In fact, the MeSH graph contains cycles, which is counterintuitive given the "more specific than" meaning of directed edges. We therefore construct a spanning DAG by suppressing edges to already touched ancestors when traversing the directed graph starting from root nodes.

[14] `http://spams-devel.gforge.inria.fr/`

Function *(M′, A) = ReassignMeshLabels(M, S, T)*

Input: MeSH thesaurus M with DAG structure, set S of data samples annotated with MeSH concepts, desired minimal number T of data samples per MeSH concept

Output: set M' of selected MeSH concepts, and an associative array A assigning each MeSH concept $m \in M'$ to a set $A(m)$ of at least T data samples

foreach *data sample* $s \in S$ **do**
  foreach *MeSH concept* m *occurring as annotation of* s **do**
    Add s to the set $B(m)$ of data samples assigned to m;
  end
end
`/* Traverse the DAG of MeSH concepts in depth-first order such that each node` m `is visited only once (stop descending as soon as a node is encountered that has already been visited) */`
foreach $m \in M$ **do**
  if $0 < |B(m)| < T$ **then**
    foreach *parent node* p *of* m **do**
      Add $B(m)$ to $B(p)$ (set union);
    end
  end
  if $|B(m)| \geq T$ **then**
    Add m to the set M' of selected MeSH concepts;
    $A(m) = B(m)$;
  end
end
return
end

Algorithm 1: MeSH label reassignment algorithm addressing the training sample size problem.

A possible limitation of this approach may arise from the dimensionality K of sparse representations and dimensionality C of concept space (number of MeSH concepts used to label data samples) needed for the MCR dataset. Experiments described in [235] were performed for $C << K$ only ($C < 40$ and $K < 400$), allowing for learning robust projections in phase (2) of the algorithm. However, first MeSH label assignment runs (see Section 4.4.1) on the MCR dataset suggest a dimensionality of $C \approx 5200$, and we expect efficiency problems with the learning algorithm when choosing K of the same magnitude. Moreover, learning linear projections into a high-dimensional concept space may easily lead to overfitting.

4.4.3 Concept Mapping

Inferring MeSH terms for a previously unseen medical case description (or case query) D using the chosen multi-view learning approach [235] requires (1) view construction for D, (2) computation of sparse representations of at least one view of D, and (3) projecting sparse representations to concept space. View construction is performed in the same manner as for dataset preprocessing (see Section 4.4.1).

Computation of optimal sparse representations, given the view dictionaries learned in the training phase, generally is an NP-hard problem, so only approximate solutions can be computed using iterative optimization [133] (provided by the SPAMS toolbox, see Section 4.4.2). The resulting computational complexity of inference is clearly a disadvantage of sparse coding techniques, and empirical results should therefore be reported after performing experiments.

On the other hand, projecting sparse representations to concept space is straightforward given the projection matrices learned during training. Projection of a single view results in a real-valued concept vector (whose length corresponds to the number of MeSH concepts used in the training phase), where weights can be interpreted as relevance values (albeit not limited to the range [0,1]). We therefore need to choose a weight threshold r, by parameter optimization on the validation set, to decide which MeSH concepts are deemed relevant and hence selected as output candidates of the concept mapping process.

Although the optimization process during multi-view learning encourages different views of one training sample to map to nearby concept vectors (according to L_2 norm), the sets of MeSH concepts produced by mapping different views of the previously unseen description D may be different. We can expect, however, that these sets will have some MeSH concepts in common, and hence propose a few alternative *concept aggregation strategies* to determine the final set of MeSH concepts assigned to D:

1. Take the union of concept sets obtained for each view.

2. Take the intersection of concept sets obtained for each view.

3. For each concept obtained for any view, compute the sum of weights for all views mapped to this concept. So if view v_i has been mapped to concept vector $(w_1^{(i)}, \ldots, w_C^{(i)})$, then the aggregated weight of concept c_k, $1 \leq k \leq C$, is computed as $w_k = \sum_{i=1}^{V} w_k^{(i)}$, where V is the number of available views. Concept c_k is selected for the final set if w_k exceeds a threshold $r_2 \geq r$, where r is the relevance threshold for candidate concepts described above. This method will prefer candidate concepts obtained for multiple views, but is not limited to the intersection. Note that this strategy generalizes strategy 1, because setting $r_2 = r$ results in selecting the union of view-specific concept sets. Strategy 2 cannot be reproduced this way, because concept weights are not limited to the range $[0, 1]$, but the intersection can be approximated by setting $r_2 = 2 * r$ in the case of two views ($V = 2$), in the sense that the final set will include the intersection. Generally, r_2 may depend on the number V of available views, e.g. $r_2 = (2 - 1/V) * r$, in order to account for the increase of the expected value of w_k for a higher number of views.

Finally, we emphasize that concept mapping is also possible for medical case descriptions or case queries that lack one of the views or even have only one view representation (e.g. textual description only), because each view is mapped to concept space separately.

4.5 Experiments

As explained in the introduction to this chapter, experimental results presented here focus on the evaluation of text-to-concept mapping approaches from the perspective of multi-label classification. More precisely, experiments measure the ability of concept mapping algorithms to reproduce manual MeSH annotations for articles of the MCR dataset. Results presented in this section were obtained in cooperation with Florian Winkler [226].

Evaluation method and performance measures are taken from text classification literature and described in Section 4.5.1. Since some of the text-to-concept mapping approaches described in Section 4.2 run into efficiency problems when applied to fulltext articles of the MCR dataset, reduced datasets had to be used, which are described in Section 4.5.2. Section 4.5.3 explains the setup and procedure used to conduct experiments, including parameter optimization. Finally, obtained results are presented in Section 4.5.4.

4.5.1 Evaluation Method

To evaluate a number of text-to-concept mapping algorithms, a dataset of text documents with ground-truth concept annotations is chosen as a test set, and each concept

mapping algorithm is applied to the test set—where ground-truth annotations are not passed as input to concept mapping—to produce a ranked list of *predicted concepts* for each test document. By comparing predicted concepts with ground-truth concepts, different evaluation measures can be applied that each produce a single number trying to capture the effectiveness of text classification achieved by a certain concept mapping algorithm on the test dataset. Since a single evaluation measure may be insufficient to compare the effectiveness of two concept mapping algorithms or to generalize results to other datasets, we use multiple measures known from text classification and information retrieval literature.

Traditionally, text classification performance is often measured by precision and recall [182]. Following a more comprehensive approach [114, 211], we distinguish three measures that compute and aggregate precision and recall values differently across all documents in the dataset: micro F_1, macro F_1, and MAP. Micro and macro F_1 are set-based measures that ignore the ranking of predicted concepts, whereas MAP represents mean average precision of ranked concept lists obtained in the same manner as for information retrieval evaluation.

To give a detailed definition of these measures, let C be the set of concepts comprising all ground-truth annotations and concepts predicted by a given concept mapping algorithm. For micro and macro measures, we further define for a given concept $c \in C$: TP_c as the number of true positives (documents for which the prediction of c is correct), FP_c as the number of false positives (documents for which the prediction of c is incorrect), and FN_c as the number of false negatives (documents for which the relevant concept c was not predicted). Both micro and macro F_1 measures aggregate precision π and recall ρ to their harmonic mean:

$$F_1 = \frac{2\,\pi\,\rho}{\pi + \rho} \tag{4.12}$$

The three measures calculate performance numbers from different perspectives of classification decisions: micro measures aggregate decisions over the entire dataset directly, macro measures first aggregate decisions on the concept level before averaging results, and MAP focuses on the document level.

Micro Measures True positives are accumulated over all concepts before computing precision and recall.

$$\pi^m = \frac{\sum_{c \in C} TP_c}{\sum_{c \in C} (TP_c + FP_c)} \tag{4.13}$$

$$\rho^m = \frac{\sum_{c \in C} TP_c}{\sum_{c \in C} (TP_c + FN_c)} \tag{4.14}$$

Macro Measures Precision and recall are determined for each concept before averaging.

$$\pi^M = \frac{1}{|C|} \sum_{c \in C} \frac{TP_c}{TP_c + FP_c} \tag{4.15}$$

$$\rho^M = \frac{1}{|C|} \sum_{c \in C} \frac{TP_c}{TP_c + FN_c} \tag{4.16}$$

Mean Average Precision Average precision AP_d is calculated for a ranked list C_d of predicted concepts for each document d in the set D of all test documents. To define average precision, we need to know which of the predicted concepts in C_d are relevant (according to ground-truth annotations); let this information be given as a boolean function R_d on concepts $c \in C$, indicating whether c is relevant for document d ($R_d(c) = 1$) or not ($R_d(c) = 0$). We then define the set P_d of rank positions of relevant concepts in C_d, the number TP_d of true positives, precision at n, and average precision for test document d by the following equations. MAP is the average of AP_d values over all documents $d \in D$.

$$P_d = \{n \mid 1 \leq n \leq |C_d|,\ R_d(C_d[n]) = 1\} \tag{4.17}$$

$$TP_d = \sum_{n=1}^{|C_d|} R_d(C_d[n]) \tag{4.18}$$

$$P@n(d) = \frac{1}{n} \sum_{i=1}^{n} R_d(C_d[n]) \tag{4.19}$$

$$AP_d = \frac{1}{TP_d} \sum_{n \in P_d} P@n(d) \tag{4.20}$$

$$MAP = \frac{1}{|D|} \sum_{d \in D} AP_d \tag{4.21}$$

All three evaluation measures described above ignore any relations between concepts that may be present in the controlled vocabulary. Because they consider the vocabulary as a "flat hierarchy" of concepts, they are also known as *flat measures*. Such evaluation measures produce conservative performance numbers in the sense that a predicted concept c is considered false positive, although there may be a ground-truth concept representing a more general or more specific notion than c. To allow for a positive contribution of such cases to precision and recall values, *hierarchical measures* have been proposed [107] that exploit the directed acyclic graph (DAG) structure of concept hierarchies. We adopt a specific hierarchical measure called LCA F_1 for experiments, whose implementation is available from the authors [107].

Table 4.4: Datasets used for text classification experiments. The *Length* column presents the average document length in words (after stop word removal), the *Concepts* column denotes the average number of ground-truth concepts per document.

Dataset	*Purpose*	*Documents*	*Length*	*Concepts*
MCR-V	validation	1000	195	12.6
MCR-V-long	validation	1000	3652	12.6
MCR-T	test	1000	197	12.6
MCR-T-long	test	1000	3709	12.6
Trieschnigg	test	1000	70.7	9.5

As a detailed definition of LCA F_1 is too involved to be presented here, we give just a general definition of set-based hierarchical measures and refer the reader to the original paper [107] for details. These measures define precision and recall using augmented sets of predicted concepts C_d and ground-truth concepts G_d for a given document d. Concept sets are augmented by sets containing lowest common ancestors (LCA) of pairs of concepts from C_d and G_d, resulting in augmented sets $C'_d \supseteq C_d$ and $G'_d \supseteq G_d$. Hierarchical precision and recall are then defined as follows, yielding the LCA F_1 measure as their harmonic mean (see (4.12)):

$$\pi^H = \frac{|C'_d \cap G'_d|}{|C'_d|} \tag{4.22}$$

$$\rho^H = \frac{|C'_d \cap G'_d|}{|G'_d|} \tag{4.23}$$

4.5.2 Datasets

For text classification experiments measuring the ability of concept mapping algorithms to reproduce manual MeSH annotations, two datasets have been used for different reasons. The first dataset, called *MCR-Random*, was generated from the ImageCLEF MCR dataset described in Section 3.1 by random sampling and should facilitate the comparison of text classification results with that of concept-based retrieval (Chapter 6) produced using the MCR dataset. The second dataset was used by Trieschnigg et al. [211] for a similar study of MeSH concept mapping algorithms, allowing us to compare our results directly to theirs[15]. Table 4.4 lists some numbers characterizing these datasets.

The *MCR-Random* dataset was created by random sampling 2000 documents from the set of 57,212 documents of the MCR dataset that are equipped with manual MeSH

[15]We thank Dolf Trieschnigg for kindly providing the dataset [211].

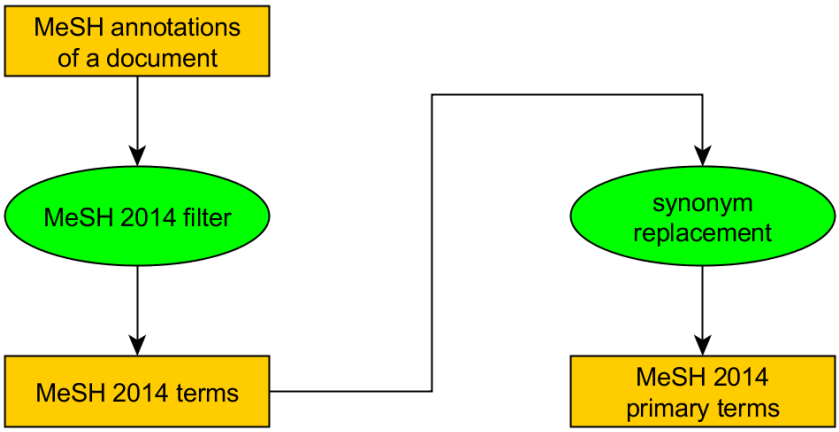


Figure 4.3: Normalization of MeSH annotations for experiments.

annotations. One half of the random subset was used for parameter optimization (*MCR-V*), the other half for evaluation of classification performance (*MCR-T*). Since not all of the evaluated text-to-concept mapping systems can cope with fulltext documents, each document in these datasets has been reduced to title and abstract only. For evaluation of capable concept mapping algorithms, the fulltext variants of these datasets (with documents including title, abstract, figure captions, and article fulltext) were retained as *MCR-V-long* and *MCR-T-long*.

The *Trieschnigg* dataset consists of 1000 MEDLINE citations containing (among other fields) title, abstract and ground-truth MeSH annotations of biomedical articles, but not the article fulltext. Compared to the *MCR-T* dataset, the average document length is much smaller (by 64%), and the average number of MeSH annotations per document is reduced by 25% (see Table 4.4). Note that the *Trieschnigg* dataset was not used for parameter tuning, but only for evaluation in our experiments.

Because the tested concept mapping systems do neither agree on the MeSH version used nor on the type of MeSH terms returned (primary term or synonym), both *MCR-Random* and *Trieschnigg* datasets underwent *MeSH normalization* before used in experiments, resulting in ground-truth MeSH annotations that consist of primary terms of the 2014 MeSH version only. MeSH normalization was performed in two steps, illustrated in Fig. 4.3:

1. *MeSH 2014 filtering:* MeSH annotations that did not correspond to a valid term (primary term or synonym) of the MeSH 2014 thesaurus were removed. Ground-truth annotations matched this MeSH version well: only 3 MeSH annotations had to be removed from the *MCR-T* dataset, 4 annotations from the *MCR-V* dataset, and none from the *Trieschnigg* dataset.

2. *Synonym replacement:* MeSH annotations referring to synonyms were replaced by their primary terms.

4.5.3 Experimental Setup

We selected six text-to-concept mapping algorithms for evaluation, representing all approaches described in Section 4.2: the existing systems *MetaMap*, *OBA*, and *MeshUp*, where the latter represents a nearest-neighbor classifier; and three of the string matching algorithms proposed in Section 4.2.3, namely *BinCov*, *Dist*, and *BinDist*. The other two string matching algorithms, *IdfBinDist* and *IdfCovDist*, were not included in experiments due to time constraints. Their evaluation can therefore be the subject of future work.

Prior to applying the selected concept mapping algorithms to the test datasets described in Section 4.5.2, parameters were optimized for each algorithm separately using a validation dataset. We used the *MCR-V* dataset for parameter optimization prior to testing on *MCR-T* and *Trieschnigg* datasets, and the *MCR-V-long* dataset for parameter tuning before testing on *MCR-T-long*.

Because *OBA* and *MeshUp* concept mapping systems do not expose any tunable parameters to the API, but use default internal settings, we focused on optimizing two thresholds meant to separate relevant from irrelevant concepts in the ranked list returned by a given concept mapping system—all selected systems assign relevance scores to returned MeSH concepts, albeit on different scales, and the result list is sorted by score in decreasing order. A *score threshold* t_s is used to declare all returned concepts with a score greater than or equal to t_s as relevant, whereas a *rank threshold* t_r is used to regard the first t_r concepts in the ranked result list as relevant. Since score and rank information can both be used to estimate actual relevance of retrieved items [229] and Trieschnigg et al. [211] used rank thresholds in their experiments, we decided to use score thresholds with the *MCR-Random* dataset and rank thresholds with the *Trieschnigg* dataset.

Additionally, the use of *specialty boosting* by string matching algorithms (see Section 4.2.3.7 on page 75) was included as a binary parameter for optimization. If set to 1, this parameter will cause string matching algorithms to assign higher scores to more special concepts (as defined by the MeSH hierarchy). All other parameters of string matching algorithms were kept at their default values (see Section 4.2.3).

Given a concept mapping algorithm, parameter optimization was performed by applying the algorithm to the validation set repeatedly with different parameter values and evaluating the concept lists returned for all documents. As score and rank thresholds do not impact the operation of concept mapping algorithms, but are applied to concept lists only, their optimization reduced to repeated evaluation of a single (long) concept list returned by the algorithm. Due to such small evaluation costs, score thresholds

Table 4.5: Text classification experiments for text-to-concept mapping algorithms.

Experiment	*Dataset*		*Algorithms*	*Threshold*
	Validation	Test		
E1	MCR-V	MCR-T	all	score
E2	MCR-V	Trieschnigg	all	rank
E3	MCR-V-long	MCR-T-long	string matching	score

Table 4.6: Optimized parameters used in text classification experiments (see Table 4.5). Entries marked by * denote situations where parameters were not applicable.

Algorithm	*Threshold*			*Specialty*		
	E1	E2	E3	E1	E2	E3
MetaMap	7.154	27	*	*	*	*
OBA	32730	29	*	*	*	*
MeshUp	0.992	17	*	*	*	*
BinCov	0.876	35	0.918	0	0	0
Dist	1.904	197	7.502	0	1	0
BinDist	1.076	33	3.422	1	1	0

were searched on a dense grid of interval length 0.001 in a range between zero and an algorithm-specific maximal value. Rank thresholds were chosen starting at 1 and incremented until classification performance started to decrease. The parameter values yielding the best classification performance with respect to a single evaluation measure were chosen as optimal values. From the various evaluation measures described in Section 4.5.1, we arbitrarily selected the *macro* F_1 measure for parameter optimization.

Using optimized parameters (score or rank thresholds), all selected concept mapping systems were applied to the *MCR-T* and *Trieschnigg* test datasets, and returned concept lists were evaluated using all four measures described in Section 4.5.1: micro F_1, macro F_1, MAP, and LCA F_1. Since *MetaMap*, *OBA*, and *MeshUp* systems could not be applied to fulltext documents for efficiency reasons, only the three selected string matching algorithms were additionally evaluated on the *MCR-T-long* dataset.

Table 4.5 summarizes the chosen configuration for text classification experiments on the three test datasets, where the *Threshold* column specifies the type of threshold optimized on the validation set. The actual values obtained by parameter optimization for each experiment are listed in Table 4.6. Note that the optimized score thresholds for string matching algorithms *Dist* and *BinDist* differ considerably between experiments E1 and E3, because concept scores produced by these algorithms depend on the document length. The *Specialty* column refers to a boolean parameter of string

Table 4.7: Classification performance of text-to-concept mapping algorithms on *MCR-T* dataset (experiment E1).

Algorithm	*micro* F_1		*macro* F_1		*MAP*		*LCA* F_1	
MetaMap	0.184		0.157		0.119		0.293	
OBA	0.174	-5.4%	0.144	-8.3%	0.093	-21.8%	0.300	+2.4%
MeshUp	0.378	+105.4%	0.241	+53.5%	0.311	+161.3%	0.534	+82.3%
BinCov	0.171	-7.1%	0.134	-14.6%	0.057	-52.1%	0.301	+2.7%
Dist	0.042	-77.2%	0.081	-48.4%	0.041	-65.5%	0.149	-49.1%
BinDist	0.184	0.0%	0.148	-5.7%	0.113	-5.0%	0.297	+1.4%

matching algorithms that determines the use of specialty boosting for concept scoring, as described earlier in this section. The inconsistent values obtained by parameter optimization across algorithms and experiments (rows and columns in the lower right quadrant of Table 4.6) are an indication of limited effectiveness of specialty boosting for experiments that aim at reproducing manual MeSH annotations.

4.5.4 Results

Evaluation results of experiment E1, which applied text-to-concept mapping algorithms on the *MCR-T* dataset of short documents, are presented in Table 4.7. In addition to numbers obtained for each evaluation measure (see Section 4.5.1), percentages denoting the change relative to MetaMap's results are given. MetaMap was chosen as a baseline, because it is used by the U.S. National Libary of Medicine to support the semi-automatic MeSH annotation of MEDLINE articles and hence represents an established text-to-concept mapping system.

The most obvious insight gained from Table 4.7 is that the nearest-neighbor classifier employed by MeshUp outperforms all other tested algorithms by large margins, consistently across all evaluation measures. This confirms similar results obtained by Trieschnigg et al. [211, 212] and can be explained by two factors contributing to effectiveness: first, nearest-neighbor classifiers utilize manual MeSH annotations of documents in the dataset, so predicted concepts are likely to match the granularity and subset of concepts[16] used for manual ground-truth annotations. Second, documents retrieved from the dataset in response to the input document are likely to have MeSH annotations that are also relevant for the input document—this is the basic assumption nearest-neighbor classifiers rely on.

[16]We hypothesize that MeSH annotations selected by human domain experts belong to a certain (domain-dependent) subset of all available MeSH concepts, and that certain levels within MeSH subtrees are preferred. This hypothesis, however, needs to be tested in future work.

Table 4.8: Classification performance of text-to-concept mapping algorithms on *Trieschnigg* dataset (experiment E2).

Algorithm	*micro* F_1		*macro* F_1		*MAP*		*LCA* F_1	
MetaMap	0.166		0.131		0.110		0.282	
OBA	0.164	-1.2%	0.131	0.0%	0.099	-10.0%	0.275	-1.5%
MeshUp	0.489	+194.6%	0.360	+174.8%	0.468	+325.5%	0.501	+77.7%
BinCov	0.100	-39.8%	0.077	-41.2%	0.081	-26.4%	0.189	-33.0%
Dist	0.033	-80.1%	0.075	-42.7%	0.056	-49.1%	0.075	-73.4%
BinDist	0.183	+10.2%	0.140	+6.9%	0.124	+12.7%	0.276	-2.1%

The Mgrep concept mapping component of OBA showed no clear advantages over MetaMap in experiment E1, although it proved to outperform MetaMap with respect to precision on several datasets using other controlled vocabularies [184, 199]. The low MAP value obtained by OBA in comparison to MetaMap (-21.8%) indicates that OBA ranked relevant concepts lower than MetaMap on average, although other measures suggest a roughly equal level of precision and recall.

Out of tested string matching algorithms, BinDist outperformed BinCov and Dist consistently across all evaluation measures other than LCA F_1. This result retrospectively justifies the design of BinDist concept scoring as a combination of BinCov and Dist, and confirms the effectiveness of *matching runs* as a design concept (see Section 4.2.3.4). An interesting insight is gained by comparing the results of BinDist and MetaMap in Table 4.7: BinDist concept mapping is roughly as effective as MetaMap on documents consisting of title and abstract, although BinDist's run-time complexity is by two orders of magnitude lower than MetaMap's.

Evaluation results on the *Trieschnigg* dataset (Table 4.8) show an even more pronounced advantage of MeshUp over all other tested concept mapping algorithms. This can be partly explained by the fact that MetaMap performed worse on this dataset than on the *MCR-T* dataset, judged on the average performance of all other algorithms. A second reason for the very good performance of MeshUp may be an overfitting effect, since the authors of MeshUp used the same dataset to evaluate their algorithm [211]; it is therefore likely that a validation set with similar characteristics has been used for parameter optimization of MeshUp.

Compared to experiment E1, the performance difference between string matching algorithms BinCov and BinDist is much larger for experiment E2. A possible explanation—whose verification would need further analysis—is that documents of the *Trieschnigg* dataset contain a higher ratio of words also found in the MeSH thesaurus than documents of the *MCR-T* dataset. The BinCov algorithm, which detects a MeSH term merely by the presence of its constituent words anywhere in the document, there-

Table 4.9: Classification performance of text-to-concept mapping algorithms on *MCR-T-long* dataset (experiment E3). Percentages denote changes relative to MetaMap results in Table 4.7.

Algorithm	*micro* F_1		*macro* F_1		*MAP*		*LCA* F_1	
BinCov	0.043	-76.6%	0.040	-74.5%	0.018	-84.9%	0.214	-27.0%
Dist	0.029	-84.2%	0.067	-57.3%	0.018	-84.9%	0.085	-71.0%
BinDist	0.150	-18.5%	0.119	-24.2%	0.078	-34.5%	0.281	-4.1%

fore detects a higher number of false positive concepts, leading to reduced performance numbers for all evaluation measures but MAP. The BinDist algorithm, on the other hand, is able to avoid most of these false positives by considering the distances of constituent words of MeSH terms within the document. This presumed property of the *Trieschnigg* dataset may also explain the low performance of MetaMap, which may have produced more false positives than BinDist due to the presence of more MeSH term words in documents.

Table 4.9 presents evaluation results of string matching algorithms on the *MCR-T-long* dataset, which was not applicable to other tested concept mapping algorithms due to efficiency problems. Percentages still refer to changes relative to MetaMap's performance on the *MCR-T* dataset (Table 4.7), because documents in both datasets share the same title and abstract.

Generally, string matching algorithms show a lower text classification performance than in experiment E1 across all evaluation measures. This behavior is expected, because longer documents give rise to more MeSH terms extracted from their content by string matching, leading to the detection of more false positive concepts with respect to manual ground-truth annotations. We note, however, that not all MeSH concepts determined as false positive in experiment E3 may actually be irrelevant due to incompleteness of manual ground-truth annotations (see footnote 1 on page 61). This statement is supported by the rather high LCA F_1 value achieved by BinDist in experiment E3 compared to MetaMap, which may be caused by predicted MeSH concepts regarded as false positive by flat measures that are actually closely related to ground-truth concepts.

Although the objective of experiments was to measure the effectiveness of text-to-concept mapping algorithms, we also recorded the execution times of algorithms for experiment E1 to provide a coarse comparison of their efficiency. Table 4.10 lists the observed mean execution times per document processed by concept mapping algorithms when applied to the *MCR-T* dataset of short documents. Time measurements for MeshUp are not available, because the MeshUp pipeline was removed from the Whatizit web service during experiments (see Section 4.2.1).

Table 4.10: Mean execution time per document for text-to-concept mapping algorithms on *MCR-T* dataset (experiment E1).

Algorithm	*Time in seconds*
MetaMap	4.28
OBA	11.8
MeshUp	N/A
BinCov	0.009
Dist	0.008
BinDist	0.010

Mean execution time of the OBA web service includes network and server latency and hence cannot be directly compared to results for other algorithms listed in Table 4.10, which were executed on a local machine. However, a mean response time of more than 11 seconds per short document substantiates that applying the OBA web service to larger document collections or longer documents will soon become impracticable. The low efficiency of MetaMap is a recognized weakness of its concept mapping approach, mainly caused by intensive use of natural language processing techniques [11]. String matching algorithms proved to be faster than MetaMap by two orders of magnitude, while providing similar or even better text classification performance in experiments E1 and E2.

4.6 Summary

This chapter described three classes of approaches that can be used to map medical case descriptions or case queries to biomedical concepts defined in controlled vocabularies or ontologies: text-to-concept mapping algorithms, image-to-concept mapping approaches, and a multi-view concept mapping approach using both textual and visual information of input documents. Since the MCR dataset comes with manual annotations referring to Medical Subject Headings (MeSH), MeSH was chosen as controlled vocabulary for experiments throughout this thesis.

The contributions of this chapter included the proposal of novel efficient text-to-concept mapping algorithms based on string matching, their experimental evaluation and comparison with existing text-to-concept mapping systems, and proposals for applying visual and multi-view concept mapping approaches to a dataset of medical case descriptions.

The effectiveness of concept mapping algorithms for retrieval will be evaluated in subsequent chapters. Therefore and due to the availability of manual ground-truth an-

notations for text documents only, experiments presented in this chapter were restricted to the evaluation of classification performance of text-to-concept mapping algorithms.

Experimental results confirmed findings already known from literature, namely that nearest-neighbor classifiers represent the most effective text-to-concept mapping approaches with respect to their ability to reproduce manual MeSH annotations. Moreover, we found that the proposed BinDist string matching approach performed as well as the MetaMap concept mapping system, which is used by the U.S. National Library of Medicine to support semi-automatic annotation of biomedical articles and citations. This is a remarkable result, because BinDist's efficiency (in terms of execution time) was observed to be higher than MetaMap's by two orders of magnitude, which makes string matching approaches a practical tool for automatic concept annotation of long documents or large document collections.

CHAPTER 5

Text-Based Retrieval

Medical case descriptions and case queries often contain informative and discriminative textual descriptions; consequently, classical text retrieval methods are an indispensable core component of medical case retrieval (MCR) algorithms and provide a solid baseline for comparisons against other methods. Based on the hypothesis (stated in Section 1.3) that the use of biomedical concepts may improve the effectiveness of MCR algorithms, this chapter investigates techniques that can be used to introduce biomedical concepts into the text retrieval process, thereby ignoring any visual information that may be present in medical case descriptions.

Text-to-concept mapping algorithms (see Section 4.2) are employed to identify biomedical concepts that are relevant for a given case query or document of the dataset, and textual representations of these concepts (MeSH terms) are added to the query or document text prior to performing text retrieval. The resulting techniques are known as *query expansion* and *document expansion* and will be described in Sections 5.1 and 5.2, respectively.

Since query expansion is performed during the online phase of MCR (cf. Fig. 1.3 on page 4), experiments focus on efficient concept mapping algorithms: *string matching* approaches (see Section 4.2.3) extract concepts already contained in the query text, and *nearest-neighbor classifiers* (see Section 4.2.2) harvest relevant concepts from pseudo-relevant documents retrieved by a separate text retrieval run. Note that string matching algorithms may add new MeSH terms to the query due to synonym replacement, but even if they do not, adding an already existing term increases its weight for text retrieval. Since the proposed usage of nearest-neighbor classifiers can be considered a query-specific local query expansion technique (see Section 2.2.3), it will also be compared to pseudo-relevance feedback methods that generate expansion terms directly from fulltext of pseudo-relevant documents.

Due to the length of fulltext documents of the MCR dataset, document expansion is performed using concept mapping algorithms based on string matching only. Section 5.3 describes the systematic evaluation of more than 500 combinations of different query and document expansion variants, leading to state-of-the-art retrieval performance on the MCR dataset without using external text corpora. Experimental results are summarized in Section 5.4, which concludes this chapter.

5.1 Query Expansion

In order to improve retrieval performance for biomedical document collections, we employ query expansion techniques utilizing two data sources for feature generation: the MeSH thesaurus, and pseudo-relevant (i.e. top-retrieved) documents. The proposed methods fall into the classes *external knowledge models* and *query-specific local techniques* described in Section 2.2.3.

There are two types of text items that can be used to expand a given query: MeSH terms and free-text terms. MeSH terms can be generated from both data sources (MeSH thesaurus and pseudo-relevant documents), whereas free-text terms are obtained from pseudo-relevant documents only. In general, any concept mapping algorithm described in Chapter 4 can be applied to a given query to obtain MeSH terms for query expansion. However, to analyze the effect of principally different techniques, we focus on three textual key methods for query expansion in our experiments: (1) MeSH terms generated by string matching from the query text, (2) MeSH terms generated from pseudo-relevant documents (kNN classifier), and (3) free-text terms generated from pseudo-relevant documents.

The following sections describe the stages of the proposed query expansion process in detail: feature generation by MeSH string matching (Section 5.1.1) and pseudo-relevance feedback (Section 5.1.2), feature selection (Section 5.1.3), and expansion term weighting (Section 5.1.4).

5.1.1 Expansion by MeSH String Matching

Using one of the MeSH string matching algorithms described in Section 4.2.3, a ranked list of MeSH terms (primary terms or synonyms) supposed to be relevant to a given query can be obtained. Since string matching algorithms ignore the synonym relationship between MeSH terms, we propose several *synonym handling methods* to determine the final list of generated features (i.e. MeSH terms):

x0 – direct No synonym handling; results of concept mapping are directly used.

x1 – primary_replace Each synonym is replaced by its corresponding primary MeSH term.

x2 – all_synonyms Each synonym is replaced by all synonyms of its corresponding MeSH record.

x3 – primary_filter Only primary MeSH terms produced by concept mapping are kept; all other MeSH terms (synonyms) are discarded. The resulting list is a filtered **direct** list.

In the final list, duplicate synonyms are suppressed, and each MeSH term receives the score of the synonym it has replaced in the original list. For example, given the query "Abdominal CT scan revealed a large left renal mass with extension into the left renal pelvis and ureter", suppose that concept mapping results in the scored list (`Ureter`: 1.0; `Pelvis, Renal`: 0.9). `Ureter` is a primary MeSH term, whereas `Pelvis, Renal` is a synonym of the primary MeSH term `Kidney Pelvis`. Here are the final lists resulting from each of the synonym handling methods described above:

x0 (`Ureter`: 1.0; `Pelvis, Renal`: 0.9)

x1 (`Ureter`: 1.0; `Kidney Pelvis`: 0.9)

x2 (`Ureter`: 1.0; `Ureters`: 1.0; `Kidney Pelvis`: 0.9; `Pelvis, Kidney`: 0.9; `Pelvis, Renal`: 0.9)

x3 (`Ureter`: 1.0)

5.1.2 Pseudo-Relevance Feedback

The second data source we used for query expansion were top-retrieved documents. The original or MeSH-expanded query is executed by a text retrieval system, and the first m documents (called *pseudo-relevant documents*) of the ranked result list are processed to generate another set of expansion features. These are added to the first query to execute the final retrieval run.

For our experiments, we used two types of expansion features generated from pseudo-relevant documents: words ranked by their TF-IDF weight in the collection, and annotated MeSH terms. In addition to single words, we also considered word n-grams (phrases of length n) ranked by TF-IDF weight. MeSH annotations are either available by manual or semi-automatic assignment—available with most PubMed publications and called *manual MeSH annotations* in the sequel—or by automatic concept mapping. More precisely, we evaluated the following expansion features generated from m pseudo-relevant documents:

r the first k words (unigrams) ranked by TF-IDF.

r2 the first k words (unigrams), and the first k_2 bigrams (word 2-grams), both ranked independently by TF-IDF.

rm all manually annotated MeSH terms.

rm2 the union of **r** and **rm** features.

raN the first k automatically annotated MeSH terms, generated and ranked by one of the string matching approaches **tN** ($1 \leq N \leq 4$) described in Section 4.2.3.

For expansion term weighting, we want all generated features to be associated with a score value. All expansion features mentioned above are already equipped with a score, except for manually annotated MeSH terms. Some of these have been marked by human annotators as *major topic*, expressing that the MeSH term represents a major topic of the document. We used this attribution to assign different scores to manually annotated MeSH terms: terms marked as *major topic* get score 1, all other MeSH terms are treated as minor topics and receive a configurable lower fixed score $s_{\min}$. We used $s_{\min} = 0.3$ in our experiments.

5.1.3 Feature Selection

The final expansion features are selected from the ranked lists generated as described in the previous sections by simple thresholds: (1) minimal concept mapping score, and (2) number of top-ranked features (parameters k and k_2 in Section 5.1.2). For selecting manually annotated MeSH terms from pseudo-relevant documents (method **rm**), we also considered reducing the set of MeSH terms to those marked as *major topic*, but that resulted in too few or even zero selected terms, because many documents of the dataset have no major topic assigned.

5.1.4 Expansion Term Weighting

The final stage of query expansion is query reformulation (see Section 2.2.3.1 on page 17). As we simply add the selected expansion features to the original query, the reformulation problem reduces to choosing expansion term weights. Because all generated features are associated with a score value, we used a variant of Rocchio's reweighting formula (see Equation (2.10) on page 18) to weight expansion terms relative to original query terms:

$$w'_{t,q'} = w_{t,q} + \mu \cdot \frac{s_t}{s_{\max}} \cdot w_{t,Q} \tag{5.1}$$

where μ is a parameter controlling the relative importance of expansion terms with respect to original query terms, and $s_{\max}$ is the maximum of expansion term scores (assumed to be positive). As in Equation (2.10), $w_{t,q}$ and $w_{t,Q}$ are the weights assigned by the underlying retrieval system to term t within the original query q and within the sequence Q of expansion terms, respectively. The normalization by $s_{\max}$ allows for unified handling of scoring functions with different scales.

Since some of the pseudo-relevance feedback methods described in Section 5.1.2 combine expansion features generated by two different scoring functions s' and s''—namely the **r2** and **rm2** methods—, we normalized their scores before applying Equation (5.1) by using a parameter κ to control the relative importance of the two scoring functions:

Table 5.1: Score thresholds of string matching algorithms used to select MeSH terms for automatic document expansion.

Document expansion	*String matching algorithm*	*Score threshold*
plus1	t1 – Dist	0.05
plus2	t2 – BinDist	0.001
plus3	t3 – IdfBinDist	0.002
plus4	t4 – IdfCovDist	0.004

$$s_t = \begin{cases} s'_t \,/\, s'_{\max} & \text{if } t \text{ was generated by } s', \\ \kappa \cdot s''_t \,/\, s''_{\max} & \text{if } t \text{ was generated by } s''. \end{cases} \tag{5.2}$$

5.2 Document Expansion

Another opportunity to address the vocabulary problem is to add terms to documents describing the topic of a document at indexing time. This may improve retrieval effectiveness if the added terms do not already occur in the original document, or occur only infrequently—provided that those terms occur in the query. This method is known as *document expansion*.

For biomedical datasets external knowledge models containing medical terms are a promising source of features for document expansion, because those terms are likely to occur in medical case queries (or in queries expressed by users). In our experiments, we expanded biomedical publications by MeSH terms supposed to capture the topic of the publication, adding these terms to the indexed *fulltext* field. In analogy to query expansion, the expansion features were identified by several methods:

- **plus** all manually annotated MeSH terms (whether marked as *major topic* or not) were used for document expansion.

- **plusN** automatically annotated MeSH terms generated by string matching algorithm **tN** described in Section 4.2.3 were used for document expansion ($1 \leq N \leq 4$). The score thresholds for MeSH term selection were determined manually by inspecting a few documents of the dataset. They are shown in Table 5.1. MeSH term matching algorithm **t0** (binary coverage) was excluded as it does not make sense for long documents.

5.3 Experiments

Sections 5.1 and 5.2 described a number of different options to implement the building blocks of query and document expansion processes that may help to improve MCR effectiveness over plain fulltext retrieval. To investigate which combinations of these implementation options actually lead to such an improvement, we conducted an experiment that systematically evaluates a large number of such combinations on the ImageCLEF MCR dataset described in Chapter 3. Details of evaluated query and document expansion methods (i.e. combinations of certain implementation options of its building blocks) are described in Section 5.3.1.

Since the effectiveness of a given query or document expansion method depends on a number of numerical parameters, a fair and meaningful comparison of different methods requires *parameter optimization* prior to final evaluation. Because the ImageCLEF MCR dataset does not contain a separate validation set for parameter optimization, we apply *cross-validation* to obtain meaningful results from partitioning the query set into validation and test subsets. Parameter optimization and the applied cross-validation methodology are described in Sections 5.3.2 and 5.3.3, respectively.

Results of cross-validation experiments are presented in Section 5.3.4. To allow for a direct comparison with MCR systems used by participants of the ImageCLEF 2013 MCR challenge [88], we additionally evaluated query and document expansion methods using the official ImageCLEF evaluation protocol. Corresponding results are presented in Section 5.3.5.

5.3.1 Evaluated Expansion Methods

The proposed query and document expansion methods described in Sections 5.1 and 5.2 are listed in Table 5.2, together with their acronyms used to identify method combinations. Note that the group *M* of MeSH SM query expansion methods is restricted to MeSH string matching (SM) algorithms described in Section 5.1.1, and pseudo-relevance feedback methods *rm* and *raN* generate MeSH expansion terms from pseudo-relevant documents and can be considered as instances of kNN concept classifiers. Due to the efficiency of MeSH string matching algorithms, methods of group *M* may also be combined with pseudo-relevance feedback methods by using a MeSH-expanded query for retrieving pseudo-relevant documents.

Every MeSH SM query expansion method uses both a MeSH string matching algorithm and a synonym handling method, amounting to $5 * 4 = 20$ query expansion methods. The other two method groups, pseudo-relevance feedback and document expansion, consist of single alternative methods, resulting in 8 and 5 methods, respectively. To compute the total number of possible method combinations, we need to take into account that every method combination must include either fulltext search or MeSH

Table 5.2: Query and document expansion methods, grouped into four *classes*. The *Count* column gives the number of different methods corresponding to each line. SM = string matching.

Acronym	*Method*	*Count*
F	*fulltext search* (no MeSH SM query expansion)	1
M	*MeSH SM query expansion*	20
tN	MeSH string matching algorithm, $0 \leq N \leq 4$	5
xN	synonym selection method, $0 \leq N \leq 3$	4
r*	*pseudo-relevance feedback*	8
r	unigrams ranked by TF-IDF	1
r2	unigrams and bigrams ranked by TF-IDF	1
rm	manually annotated MeSH terms	1
rm2	union of r and rm features	1
raN	automatically annotated MeSH terms ranked by score tN, $1 \leq N \leq 4$	4
+*	*document expansion*	5
+	manually annotated MeSH terms	1
+N	automatically annotated MeSH terms ranked by score tN, $1 \leq N \leq 4$	4

SM query expansion ($1 + 20 = 21$ possibilities), and that a pseudo-relevance feedback or document expansion method may be used or not (resulting in $8 + 1 = 9$ and $5 + 1 = 6$ possibilities, respectively). Thus, the total number of proposed query and document expansion method combinations is $21 * 9 * 6 = 1134$.

To reduce overall computation time[1] and to simplify analysis and presentation of results, we chose to evaluate only "interesting" method combinations, not all possible ones. Preliminary experiments showed that methods employing pseudo-relevance feedback gave clearly better results than other method combinations, so we emphasized feedback methods when selecting combinations for evaluation. Moreover, we were interested in MeSH query expansion alone, and in combinations of document expansion with feedback methods. The selected set of 546 method combinations is presented in Table 5.3, grouped by combinations of three classes of techniques: MeSH SM query expansion (M), pseudo-relevance feedback (r*), and document expansion (+*). The acronym *raN+N* denotes all method combinations using pseudo-relevance feedback of automatically annotated MeSH terms (*raN*) and document expansion (*+N*) using the *same* MeSH string matching method N ($1 \leq N \leq 4$) (cf. Table 5.2). We assume

[1]Evaluating 546 method combinations concurrently on a 24-core machine with 96 GB RAM took about 36 hours.

Table 5.3: Query and document expansion methods selected for evaluation.

Acronym	*Group of methods*	*Count*
F	fulltext search (without query expansion)	1
M	MeSH SM query expansion	20
F+	fulltext search with document expansion (manual MeSH annotation)	1
M+	MeSH SM query expansion with document expansion (manual MeSH annotation)	20
Fr*	fulltext search with pseudo-relevance feedback	8
Mr*	MeSH SM query expansion followed by pseudo-relevance feedback	160
Fr*+*	fulltext search with pseudo-relevance feedback and document expansion Fr+, Frm+, FraN+N, Frm2+*, Fr2+*	16
Mr*+*	MeSH SM query expansion followed by pseudo-relevance feedback with document expansion Mr+, Mrm+, MraN+N, Mrm2+*, Mr2+*	320
Total count		546

that these combinations perform better than cross-combinations *raN+K* with $N \neq K$, because MeSH terms chosen for query expansion from pseudo-relevant documents are more likely to be found in expanded documents if MeSH terms of both expansions have been generated by the same algorithm.

5.3.2 Parameter Optimization

The query expansion methods described in Section 5.1 introduce a number of free parameters that need to be chosen carefully to optimize retrieval performance on a given dataset. As there are many combinations of methods to be evaluated and optimal parameter settings are sensitive to the particular method combination in use, an automatic parameter optimization algorithm was applied. Moreover, the use of automatic parameter optimization facilitates evaluation in a cross-validation setting, where only part of the dataset is used to optimize parameters and the remaining part is used to assess retrieval performance.

The parameters to be optimized for query expansion methods are listed in Table 5.4. Note that not all parameters are relevant for every expansion method. For example, expansion method *Frm* (expansion with manually annotated MeSH terms taken from pseudo-relevant documents, see Table 5.2) takes only parameters m, k, and μ_F. Table 5.5 lists the number of relevant parameters for the query and document expansion

Table 5.4: Parameters to be optimized for query expansion methods described in Section 5.1.

Parameter	*Type*	*Range*	*Description*
$s_{\min}$	real	0.2 – 2.0	minimal matching score for MeSH SM term selection
μ_M	real	0.1 – 1.0	weighting factor of MeSH SM expansion terms relative to original query terms
m	integer	1 – 20	number of pseudo-relevant documents
k	integer	1 – 150	number of expansion terms to use for pseudo-relevance feedback
k_2	integer	1 – 50	number of bigrams to use for *r2* expansion method
μ_F	real	0.1 – 2.0	weighting factor of feedback terms relative to original query terms
κ	real	0.1 – 2.0	relative importance of the two scoring functions for *r2* and *rm2* methods

Table 5.5: Number of parameters to be optimized for query and document expansion methods used in experiments. Acronyms of methods are defined in Table 5.2.

Combinations	*Parameters*	*Count*
F, F+	0	2
M, M+	2	40
Fr, Frm, FraN, Fr+, Frm+, FraN+N	3	12
Frm2, Frm2+*	4	6
Fr2, Fr2+*	5	6
Mr, Mrm, MraN, Mr+, Mrm+, MraN+N	5	240
Mrm2, Mrm2+*	6	120
Mr2, Mr2+*	7	120
Total count		546

methods used in experiments. There are two method groups with 5 parameters, because they use different parameter sets.

As objective function to be maximized we use mean average precision (MAP) of a retrieval run on the validation dataset. Because evaluation of the objective function at a single point in parameter space is a costly operation, we chose an optimization algorithm that tries to keep the number of objective function evaluations low: Simultaneous Perturbation Stochastic Approximation (SPSA) [196, 197]. It has been designed to find a local optimum of continuous-variable problems with smooth objective functions, even

Table 5.6: Statistics of applying SPSA to parameter optimization during 5-fold cross-validation of 546 retrieval configurations. The total number of optimization runs is $5 * 546 = 2730$.

Number of optimization runs	2730	100%
Number of improved runs	2424	89%
Number of converged runs	635	23%
Number of runs yielding optimum in last iteration	270	10%

if objective function measurements include added noise.

Although sufficient conditions for convergence of SPSA cannot be established for our parameter optimization problem—some parameters take discrete values, and the objective function is not continuous—, we can use SPSA as a vehicle for heuristic optimization of parameters: the algorithm performs a "random walk" in parameter space guided by objective function differences, and we consider the best of visited points as an "optimal" parameter setting. By choosing manually tuned parameter settings as a starting point, we ensure that the result of parameter optimization will not be worse than a previously known "best" parameter configuration. The usefulness of this heuristic application of SPSA becomes evident after the fact when looking at some statistical results of parameter optimization during 5-fold cross-validation of 546 retrieval configurations, given in Table 5.6. In 89% of optimization runs, SPSA found better parameter settings, although only 10% of optimizations obtained the best setting in the last iteration (no matter whether SPSA converged or not).

The SPSA algorithm is easy to implement and is shown in Fig. 5.1. It is formulated to minimize a *loss function* y by finding an optimal value of p-dimensional vector $\vec{\theta}$, which is produced as output at the end of the presented MATLAB code. Starting with an initial guess $\vec{\theta}_1$ and non-negative parameters a, c, A, α, and γ, each iteration k computes an approximation $\vec{g}_k$ of the unknown gradient of y at $\vec{\theta}_k$. The gradient computation (5.5) requires only two evaluations of the loss function at points $\vec{\theta}_k^+$ and $\vec{\theta}_k^-$ according to Equations (5.3) and (5.4). (c_k) is a decreasing sequence of positive numbers and $\vec{\Delta}_k$ is a random perturbation vector whose elements are ± 1, sampled independently from a Bernoulli distribution with probability 1/2. $\vec{\theta}_k$ is then updated to a new value $\vec{\theta}_{k+1}$ (supposed to be closer to the minimum) by adding the negative gradient approximation scaled by a positive number a_k that decreases with k (5.6).

$$\vec{\theta}_k^+ = \vec{\theta}_k + c_k \vec{\Delta}_k \tag{5.3}$$

$$\vec{\theta}_k^- = \vec{\theta}_k - c_k \vec{\Delta}_k \tag{5.4}$$

```
For k = 1:n
        ak = a/(k+A)^alpha;
        ck = c/k^gamma;
        delta = 2*round(rand(p,1)) - 1;
        thetaplus = theta + ck*delta;
        thetaminus = theta - ck*delta;
        yplus = loss(thetaplus);
        yminus = loss(thetaminus);
        g = (yplus - yminus) ./ (2*ck*delta);
        theta = theta - ak*g;
        theta = min(theta, thetamin);
        theta = max(theta, thetamax);
end
theta
```

Figure 5.1: MATLAB code of SPSA algorithm [197]. Initialization and stopping criterion are not shown.

$$\vec{g}_k = \frac{y(\vec{\theta}_k^+) - y(\vec{\theta}_k^-)}{2\,c_k} \begin{bmatrix} \Delta_{k1}^{-1} \\ \Delta_{k2}^{-1} \\ \vdots \\ \Delta_{kp}^{-1} \end{bmatrix} \tag{5.5}$$

$$\vec{\theta}_{k+1} = \vec{\theta}_k - a_k\,\vec{g}_k \tag{5.6}$$

To apply the SPSA algorithm to parameter optimization for query expansion we normalized every parameter domain to the interval $[0, 1]$ by linear transformation and used negative MAP of retrieval runs as loss function. Prior to evaluating the loss function, the inverse linear transform needs to be applied to normalized parameter values, followed by rounding for originally integer-valued parameters. Normalized parameter values were clipped to the $[0, 1]$ range when applying the update step (5.6). The algorithm terminates when $y(\vec{\theta}_k^+)$ and $y(\vec{\theta}_k^-)$ differ by less than ε for K successive iterations, or when a maximal iteration count n is reached. SPSA parameter values used in experiments are shown in Table 5.7.

To determine the result $\vec{\theta}_{\min}$ of optimization we consider all parameter vectors $\vec{\theta}_k^+$ and $\vec{\theta}_k^-$ as well as the initial vector $\vec{\theta}_1$ and the final vector $\vec{\theta}_{n+1}$ when the algorithm terminates after n iterations. The most recently computed one of these parameter vectors with minimal loss value is selected as $\vec{\theta}_{\min}$.

Table 5.7: SPSA parameters used in experiments.

Parameter	*Value*	*Description*
a	1.0	used to compute a_k
A	0	used to compute a_k
α	1.0	used to compute a_k
c	0.1	used to compute c_k
γ	0.5	used to compute c_k
ε	0.001	equality threshold for stopping criterion
K	3	number of stationary iterations for stopping criterion
n	20	maximal iteration count

$$\vec{\theta}_{\min} = \underset{1 \le k \le n}{\operatorname{argmin}} \left\{ y(\vec{\theta}_1),\ y(\vec{\theta}_k^+),\ y(\vec{\theta}_k^-),\ y(\vec{\theta}_{n+1}) \right\} \tag{5.7}$$

5.3.3 Cross-Validation

When applying parameter optimization prior to evaluating a given retrieval algorithm, it is important to use different datasets for parameter optimization and evaluation to avoid sacrificing the generalization power of evaluation results. This is ideally achieved by using a separate *validation* dataset for parameter optimization that is disjoint from the *test* dataset used to evaluate the algorithm. If such separate datasets are not available, the *cross-validation* methodology (see e.g. [25, 182]) can be applied to achieve generalizable results even from a single dataset.

The general idea of *n-fold cross-validation* is to split the dataset into n partitions of roughly equal sizes and use the union of $n-1$ partitions for parameter optimization and the remaining partition for testing. To compensate for the small size of the test partition, parameter optimization and testing are repeated with a different selection of $n-1$ validation partitions and one test partition, until each of the n partitions has been used once for testing. Finally, the average performance over all n evaluation runs is reported as system performance of the evaluated algorithm.

Partitioning the dataset is a clear task in the context of machine learning, where cross-validation has its origin, but it needs a different interpretation in the context of information retrieval, where a dataset consists of a document collection and a set of queries with corresponding relevance judgments. Here, partitioning needs to be applied to the query set, because queries are the units for which retrieval performance is measured. Partitioning the document collection, on the other hand, does not provide any benefits for separating concerns of parameter optimization and testing. In fact, reducing the size of the document collection would have undesirable effects on measurement of

retrieval performance: the search space gets smaller, and some judged documents (even relevant documents) may have been removed.

Since the ImageCLEF MCR dataset described in Chapter 3 does not provide disjoint datasets for validation and testing[2], and no other MCR dataset was available to us, we applied 5-fold cross-validation to use the given dataset for both parameter optimization and testing. The set of 35 queries was partitioned into 5 subsets of equal sizes, and the average of five MAP values measured on the five query subsets (after separate parameter optimization) is reported as performance measure for a given MCR algorithm.

To illustrate the cost generated by this evaluation method, we determine the maximal number of retrieval runs needed to compute the final MAP value of a given MCR algorithm. For 5-fold cross-validation, parameter optimization is applied to each of five validation sets consisting of 28 queries, and retrieval performance is measured on each of five test sets comprising 7 queries. An optimization run is limited to 20 iterations, each computing MAP on 28 queries twice (with different parameter settings, see Section 5.3.2), amounting to $20 * 28 * 2 = 1120$ retrieval runs at most. Evaluation on the test set requires 7 retrieval runs. We end up with a maximum of $5 * (1120 + 7) = 5635$ retrieval runs to evaluate a single MCR algorithm. Note that this number does not include additional "internal" retrieval runs executed during pseudo-relevance feedback for some of the methods described in Section 5.3.1.

5.3.4 Cross-Validation Results

We evaluated the selected 546 combinations of query and document expansion methods by 5-fold cross-validation on the ImageCLEF MCR dataset, as explained in the previous sections. As retrieval performance metric we used *mean average precision* (MAP), which is commonly applied to TREC-style evaluations (see Section 2.2.2). Note that the same metric served as objective function for parameter optimization (Section 5.3.2).

Figure 5.2 presents a scatter plot of obtained results, grouped by the eight classes of method combinations listed in Table 5.3. Every data point represents the MAP value of a method combination obtained by cross-validation.

The two best method combinations of each group are listed in Table 5.8, revealing the actual algorithms employed. In particular, the overall best method combination was Mt2x0r2, which used MeSH term matching algorithm **BinDist** (t2) with direct synonym handling (x0) for MeSH query expansion, followed by pseudo-relevance feedback with unigrams and bigrams (r2) to further expand the query. Refer to Table 5.2 and Section 5.3.1 to interpret acronyms of other method combinations.

[2]Earlier editions of the ImageCLEF 2013 MCR dataset are subsets containing a reduced document collection or query set. The ImageCLEF 2012 MCR dataset, for example, contains the same document collection and a subset of 26 out of 35 queries of the 2013 edition.

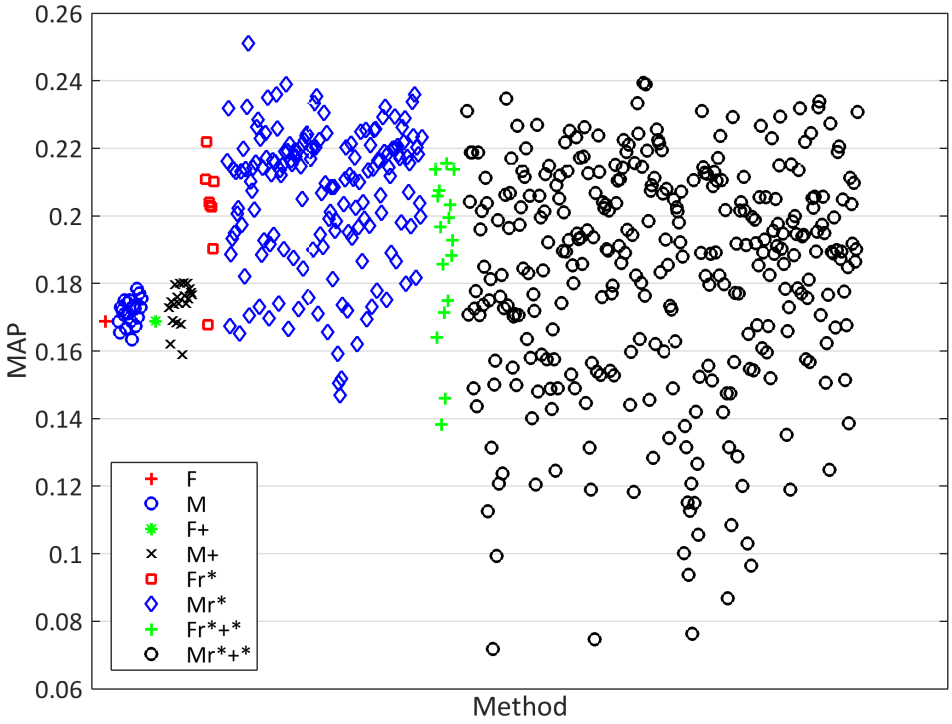


Figure 5.2: Scatter plot of 546 combinations of query and document expansion methods with optimized parameters obtained by 5-fold cross validation on the ImageCLEF 2013 MCR dataset. Method combinations are grouped according to Table 5.3.

Table 5.8: Best and second-to-best combinations of query and document expansion methods depicted in Figure 5.2. Best MAP values of each column are marked in boldface.

Group	*Best Method*	*MAP*	*Second Method*	*MAP*
F	F	0.1689	–	–
M	Mt0x3	0.1784	Mt2x3	0.1771
F+	F+	0.1688	–	–
M+*	Mt2x2+	0.1802	Mt0x3+	0.1801
Fr*	Fr2	0.2219	Fr	0.2109
Mr*	Mt2x0r2	**0.2511**	Mt1x1r	**0.2390**
Fr*+*	Frm2+	0.2155	Fr2+	0.2139
Mr*+*	Mt4x1r2+	0.2393	Mt4x1r2+2	0.2389

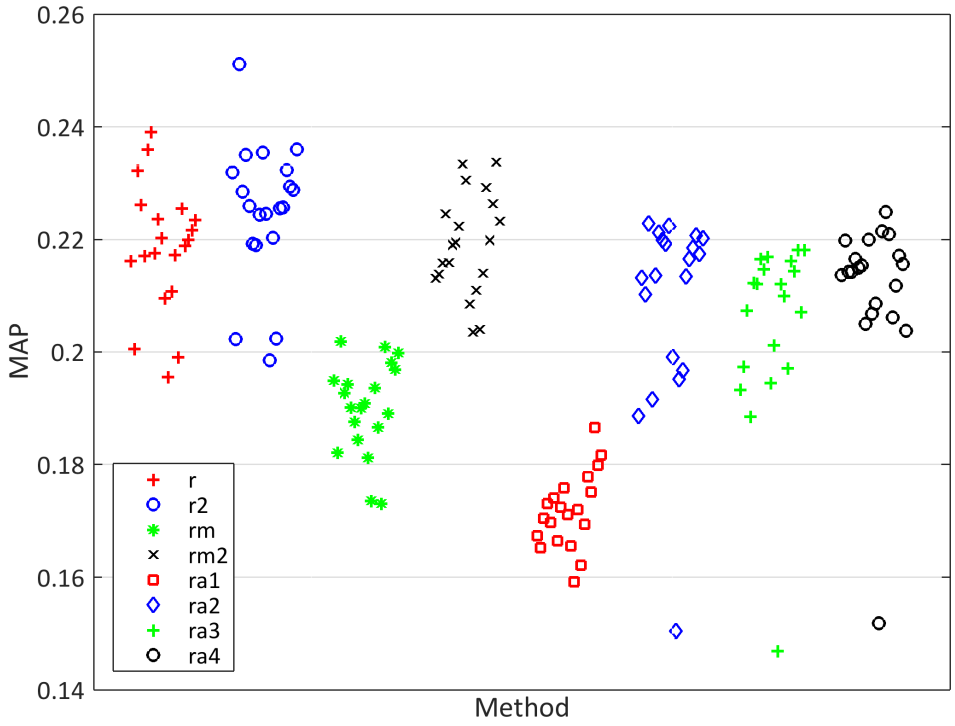


Figure 5.3: Scatter plot of 160 query expansion methods employing MeSH query expansion followed by pseudo-relevance feedback, grouped by feedback method. The data points correspond to the Mr* group of Figure 5.2. Acronyms of feedback methods are explained in Table 5.2.

The following three sections analyze results for pseudo-relevance feedback methods, query expansion by MeSH string matching, and document expansion in more detail.

5.3.4.1 Comparison of Feedback Methods

As all method combinations exceeding 0.2 MAP employ pseudo-relevance feedback, we would like to know if some feedback methods are consistently better than others within a given group of combinations. We focused on the best performing group Mr* and grouped their methods by employed pseudo-relevance feedback algorithm. The scatter plot (Figure 5.3) reveals that point clouds pertaining to different feedback algorithms form clusters with rather small intra-class variance (with respect to MAP), and some classes clearly perform better than others, indicated by large inter-class distances. In particular, feedback methods ranking unigrams (words) of pseudo-relevant documents by TF-IDF, namely methods r, r2, and rm2, perform consistently better than other

feedback methods. Although the overall best method combination uses unigrams and bigrams for feedback (r2), this feedback method cannot be claimed to be better than feedback using unigrams only (r), because the best data point appears to be an outlier in the group of tested r2 method combinations.

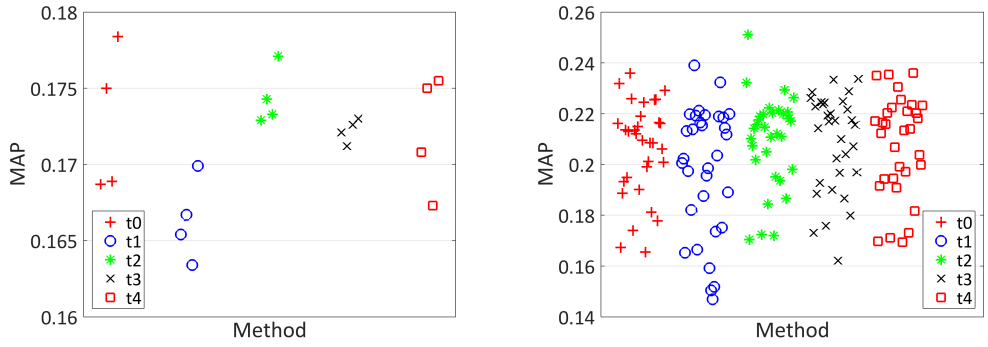
Another interesting conclusion drawn from Figure 5.3 is that method combinations using manually annotated MeSH terms for feedback (rm) consistently perform worse than feedback methods ra2, ra3, and ra4, which all use automatically annotated MeSH terms for feedback. This may be unexpected to some extent, because manually annotated MeSH terms are assumed to be more accurately related to document semantics than automatically annotated ones and hence should provide a more effective data source for query expansion. But in the light of successful feedback methods using words from pseudo-relevant documents directly (r, r2, and rm2 methods), the relatively better performance of ra2, ra3, and ra4 methods becomes intelligible, as they basically extract MeSH terms already present in documents.

Method combinations employing feedback by MeSH terms extracted using distance-based match frequency (ra1, see **Dist** MeSH string matching in Section 4.2.3) perform consistently worse than ra2, ra3, and ra4 methods. This is a strong indication that the concept of matching runs (used by ra2, ra3, and ra4 methods) is important to apply the proposed MeSH string matching approach to longer documents. The **Dist** scoring function may assign a high score to a MeSH term for a document just because words of the MeSH term occur sufficiently often in the document, although not all words of the MeSH term are present or they occur in distant locations in the document.

5.3.4.2 MeSH SM Query Expansion Methods

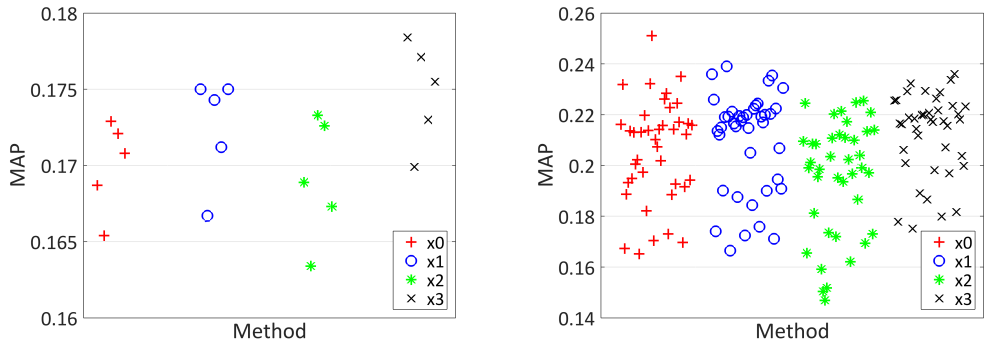
The 20 data points of group M in Figure 5.2 suggest that the effectiveness of query expansion methods employing MeSH string matching (SM) algorithms (see Section 4.2.3) is small. The comparison of different MeSH string matching algorithms (*t0* – *t4*)and synonym handling methods (*x0* – *x3*) is therefore likely to give no clear results, but is pursued here in the interest of completeness.

Figure 5.4 presents scatterplots of method combinations M (MeSH SM query expansion) and Mr* (MeSH SM query expansion followed by pseudo-relevance feedback), grouped by MeSH string matching algorithms. Although the plot for group M (Fig. 5.4(a)) suggests that the **BinDist** algorithm (t2) performs better than **Dist** (t1), the difference in terms of MAP is small enough to be swallowed by the dominating variance of feedback methods in group Mr* (Fig. 5.4(b)). A similar observation can be made for scatterplots grouped by MeSH synonym handling methods (Fig. 5.5).



(a) (b)

Figure 5.4: Scatter plot of MeSH SM query expansion methods, grouped by MeSH string matching algorithm. The data points correspond to (a) group M and (b) group Mr* of Figure 5.2.



(a) (b)

Figure 5.5: Scatter plot of MeSH SM query expansion methods, grouped by MeSH synonym handling method. The data points correspond to (a) group M and (b) group Mr* of Figure 5.2.

5.3.4.3 Document Expansion Methods

When comparing the point clouds of method groups Mr* and Mr*+* in Figure 5.2, it is obvious that document expansion (employed by Mr*+* methods) did not improve retrieval performance over query expansion methods (Mr*). It even deteriorated results substantially for many method combinations. However, for the sake of comparing the usefulness of automatic MeSH annotation algorithms based on string matching with that of manual MeSH annotations, it may be interesting to take a closer look at the effectiveness of different tested document expansion methods (see Section 5.2).

Figure 5.6 presents all data points corresponding to query expansion methods us-

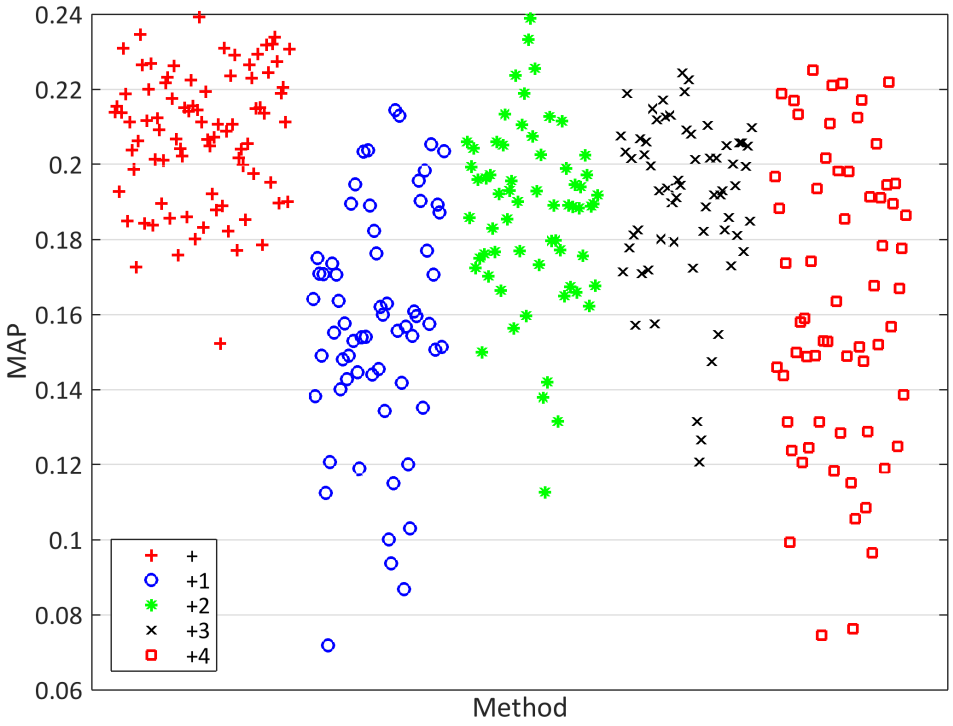


Figure 5.6: Scatter plot of query expansion methods using pseudo-relevance feedback combined with document expansion, grouped by document expansion method. The data points correspond to groups Fr*+* and Mr*+* of Figure 5.2.

ing pseudo-relevance feedback combined with document expansion (groups Fr*+* and Mr*+* of Figure 5.2), grouped by document expansion method. In contrast to their use for pseudo-relevance feedback (see Section 5.3.4.1), manually annotated MeSH terms perform consistently better than automatically annotated ones for document expansion. This can be explained by the fact that manual MeSH annotations are more likely to add relevant, not already existing information to a document than automatically annotated MeSH terms generated by string matching.

However, MeSH terms annotated by the **BinDist** algorithm (t2) yield a remarkable performance for many method combinations that is comparable to document expansion with manual MeSH annotations, including the top-performing ones. On the other hand, the more sophisticated MeSH string matching algorithms **IdfBinDist** (t3) and **IdfCovDist** (t4) were not as effective as **BinDist** (t2), which suggests that IDF weighting should not be used with MeSH string matching for document expansion.

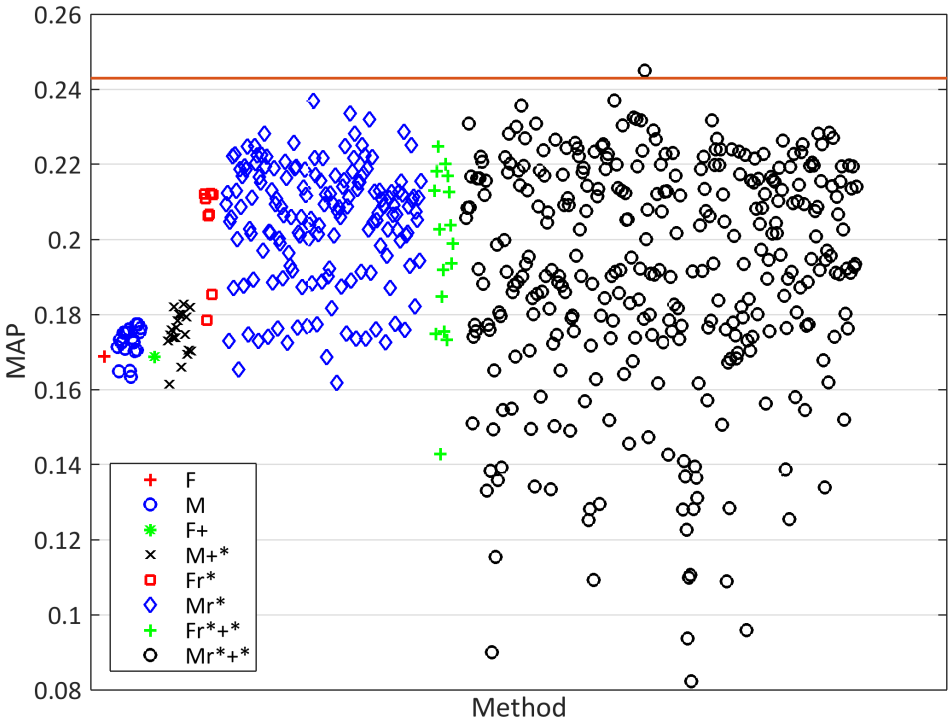


Figure 5.7: Scatter plot of 546 combinations of query and document expansion methods with parameters optimized on corrected ImageCLEF 2012 dataset and evaluated on the ImageCLEF 2013 MCR dataset. The horizontal line at MAP 0.2429 corresponds to the best run submitted to ImageCLEF 2013 [88].

5.3.5 ImageCLEF Evaluation Results

In order to compare the proposed methods to retrieval runs originally submitted to the ImageCLEF 2013 MCR task [88], we additionally evaluated the proposed query and document expansion methods using the official ImageCLEF 2013 MCR evaluation protocol. To simulate information available to participants of ImageCLEF 2013 before submitting their results, we used the ImageCLEF 2012 MCR dataset for parameter optimization. Retrieval performance was then evaluated for each optimized method combination on the ImageCLEF 2013 MCR dataset.

The ImageCLEF 2012 MCR dataset consists of the same document collection as the 2013 dataset (see Section 3.1), but provides only 26 queries that form a subset of the 35 queries contained in the 2013 dataset and, most importantly, uses a different document pool for relevance judgments than the 2013 dataset. That is, for a given query occurring in both datasets, the sets of judged documents (marked as relevant or

Table 5.9: Best and second-to-best combinations of query and document expansion methods, optimized on corrected ImageCLEF 2012 dataset and evaluated on ImageCLEF 2013 MCR dataset. Best MAP values of each column are marked in boldface.

Group	*Best Method*	*MAP*	*Second Method*	*MAP*
F	F	0.1689	–	–
M	Mt0x3	0.1774	Mt2x3	0.1774
F+	F+	0.1688	–	–
M+*	Mt3x2+	0.1827	Mt0x1+	0.1820
Fr*	Fra4	0.2122	Fr	0.2121
Mr*	Mt3x1rm2	0.2369	Mt2x2ra4	0.2335
Fr*+*	Fr2+3	0.2247	Frm2+	0.2201
Mr*+*	Mt4x1r2+2	**0.2450**	Mt2x1rm2+	**0.2370**

non-relevant) are not the same, although unlikely to be disjoint. In particular, the 2012 relevance judgments failed to provide any relevant documents for three queries, which makes these queries useless for evaluation, because average precision would always be zero. We therefore removed these queries from the 2012 dataset and used the corrected dataset with 23 queries for parameter optimization.

We emphasize, however, that this evaluation method is susceptible to overfitting and hence provides only limited generalization power, because 2/3 of the query set used for testing is also employed for parameter optimization, even if relevance judgments used for validation and testing are not exactly equal. But since participants of ImageCLEF 2013 are likely to have used the 2012 dataset for parameter optimization, a fair comparison of proposed MCR methods with their systems should use the same evaluation method.

In analogy to Section 5.3.4, a scatter plot of all 546 tested method combinations is shown in Fig. 5.7. The details of the best two method combinations of each method group are listed in Table 5.9. The best MCR run submitted to ImageCLEF 2013 achieved 0.2429 MAP using an external corpus of 22 million MEDLINE[3] citations to generate MeSH terms for query expansion by local feedback [46]. This run is indicated by a horizontal line in Fig. 5.7. Although all our methods rely on the dataset corpus only, one method combination (Mt4x1r2+2) achieved an even better result; it employs query expansion by MeSH terms extracted from the query using **IdfCovDist** string matching (t4) and synonym replacement (x1), followed by pseudo-relevance feedback using TFIDF-weighted unigrams and bigrams (r2), and document expansion with MeSH terms extracted by **BinDist** string matching (+2).

When comparing the scatter plots obtained by cross-validation (Fig. 5.2) and ImageCLEF evaluation (Fig. 5.7), they give a similar picture of effectiveness of differ-

[3] `https://www.nlm.nih.gov/pubs/factsheets/medline.html`

ent method groups. Even the absolute MAP values achieved by the vast majority of method combinations within a group coincide; in particular, most Mr* methods achieve a MAP between 0.16 and 0.24 with both evaluation methods). Outliers, however, both in high- and low-performing ranges, differ remarkably in several method groups (Fr*, Mr*, and Mr*+*). We attribute that primarily to randomness inherent to parameter optimization, but also—for high-performing outliers in Fig. 5.7—to overfitting caused by parameter optimization on the ImageCLEF 2012 dataset.

Based on the correspondence between ImageCLEF-type evaluation and cross-validation, the main findings of Section 5.3.4 remain valid, and we do not repeat the analysis here. In particular, query expansion methods employing MeSH SM query expansion followed by pseudo-relevance feedback (group Mr*) seem to be the best choice, and combining them with document expansion (group Mr*+*) has no further benefit.

5.4 Summary

This chapter investigated the benefit of selected query expansion and document expansion techniques to textual methods for medical case retrieval (MCR). We used the string matching approaches proposed in Chapter 4 to automatically map queries or documents to Medical Subject Headings (MeSH), and used these MeSH terms for query or document expansion. Additionally, query-specific local feedback methods were used to determine expansion terms from top-retrieved documents. Several variants of these query and document expansion methods were combined and evaluated on the ImageCLEF 2013 MCR dataset described in Chapter 3. More precisely, 546 method combinations were evaluated independently by 5-fold cross-validation to avoid overfitting by parameter optimization. Another set of experiments applied the official ImageCLEF 2013 MCR evaluation procedure to these method combinations to allow for comparison with retrieval runs submitted to ImageCLEF 2013.

Experimental results show that query expansion methods using MeSH terms produced by string matching (SM) and local feedback can substantially improve MCR performance over fulltext-only retrieval and achieve state-of-the-art retrieval performance on the ImageCLEF 2013 MCR dataset. The improvement is mainly due to local feedback using unigrams (single words) and bigrams (2-word sequences) from pseudo-relevant documents, local feedback by MeSH terms is less effective. However, combining MeSH SM query expansion with local feedback may result in a higher performance gain (in terms of mean average precision) than combining it with fulltext-only retrieval.

On the other hand, combining MeSH SM query expansion and/or local feedback with document expansion does not improve retrieval performance. There is no consistent best method within the set of proposed MeSH string matching algorithms and MeSH synonym handling methods used for query and document expansion.

Although care has been taken to avoid overfitting effects when performing experiments, the generalization power of results is still limited by the facts that (1) evaluation is based on a single dataset, and (2) results depend on the effectiveness of parameter optimization. So further work could improve evaluation by searching for or developing a second dataset, and by cross-validating parameter optimization using a different (e.g. genetic) algorithm. Other promising avenues for future work on textual MCR techniques include utilizing document structure (title, abstract, image captions), applying more sophisticated query expansion methods (cf. Section 2.2.3), or using external corpora or text categorization based on machine learning [182] to expand queries or annotate documents with additional biomedical terms.

CHAPTER 6

Concept-Based Retrieval

This chapter introduces a principled approach to utilizing biomedical concepts for medical case retrieval (MCR). The applied method follows the paradigm of concept-based retrieval [193, 212] and, therefore, differs from the textual query expansion and document expansion methods described in Chapter 5. The general idea of concept-based retrieval is to represent a document or query by a vector of relevance scores identifying relevant concepts, called *concept vector*, and implementing ranking and retrieval by a similarity measure on concept vectors. The resulting retrieval model can be considered as an instance of the vector space model of information retrieval (IR, see Section 2.2.1) and can therefore be implemented using existing text retrieval technology and software. Details of the concept-based retrieval method applied to MCR are described in Section 6.1.

To evaluate concept-based retrieval, we apply concept mapping algorithms (see Chapter 4) to documents and queries of the MCR dataset and measure the resulting concept-based retrieval performance. Corresponding results are presented in Section 6.2. Since concept-based retrieval is known to be less effective than fulltext retrieval on biomedical text corpora [212], evaluation results of this chapter serve other purposes: they provide another and—for the purpose of MCR—more relevant evaluation criterion for concept mapping algorithms than text classification results presented in Chapter 4; and they allow to assess the contribution of the concept-based retrieval component in multimodal approaches described in Chapter 7. Moreover, results are compared to an "ideal" concept-based retrieval method that uses concept annotations of actually relevant documents (according to ground-truth judgments) to represent queries. Results and findings are summarized in Section 6.3, which concludes the chapter.

6.1 Applied Method

Since concept-based retrieval follows the vector space model of text retrieval, we used the Lucene text retrieval engine (see Section 2.2.1) to index and retrieve biomedical

documents by their concept vectors. The role of indexed terms in text documents is taken by unique textual representations of MeSH concepts, realized by MeSH node identifiers like C13.703.039 (see Section 4.1). If a single MeSH concept relevant for a given document or query is equipped with multiple node identifiers—which means that the MeSH concept is a member of multiple trees in the MeSH hierarchy—, then all these node identifiers are included in the concept vector of the document or query.

We note that Lucene does not allow to specify the weight of a term within the term vector representation of a document explicitly, because the TF-IDF weighting scheme employed by Lucene derives term weights from the number of occurrences of the term within the document (term frequency, TF) and within the document collection (inverse document frequency, IDF). MeSH node identifiers assigned to a document or query (by manual annotation or automatic concept mapping) will therefore be represented by Lucene with a weight derived from "term frequency" 1, because a MeSH concept is assigned only once to the document or query.

Since concept weights cannot be changed easily for documents indexed by Lucene, we looked for a way to adapt the weights of query concepts according to the reliability of MeSH annotations of indexed documents (manual annotations are considered more reliable than automatic ones). We chose to use three separate Lucene *index fields* for different types of MeSH annotations of documents, namely for major manual annotations, minor manual annotations, and automatic annotations (see Section 4.2.2). Index fields are indexed and searched separately by Lucene, and Lucene's query syntax and default scoring function (see Section 2.2.1) allow to specify field-specific weights (boosting factors) for query terms.

During concept-based indexing of the MCR dataset, these three types of MeSH annotations of a document are transformed to MeSH node identifiers and indexed in three corresponding fields by Lucene. Manual annotations are available with the majority of documents in the MCR dataset (see Section 3.1), whereas automatic annotations are created on the fly from fulltext articles using the BinDist string matching algorithm—which proved to be the most effective text-to-concept mapping approach that is efficient enough to process large collections of long documents, as recognized in Chapter 4. Note that the indexing process ignores article images contained in documents of the MCR dataset, because separate manual MeSH annotations are not available for images and image captions are already included in the fulltext of documents.

Figure 6.1 illustrates the concept-based retrieval process and displays a schematic representation of the concept-document index created by Lucene, where MeSH node identifiers have been replaced by primary MeSH terms for easier comprehension. The inverted index built from MeSH-annotated documents of the MCR dataset is searchable by MeSH concepts and, for each MeSH concept and index field, stores a list of references to documents annotated with this concept (and additional TF-IDF weights not shown

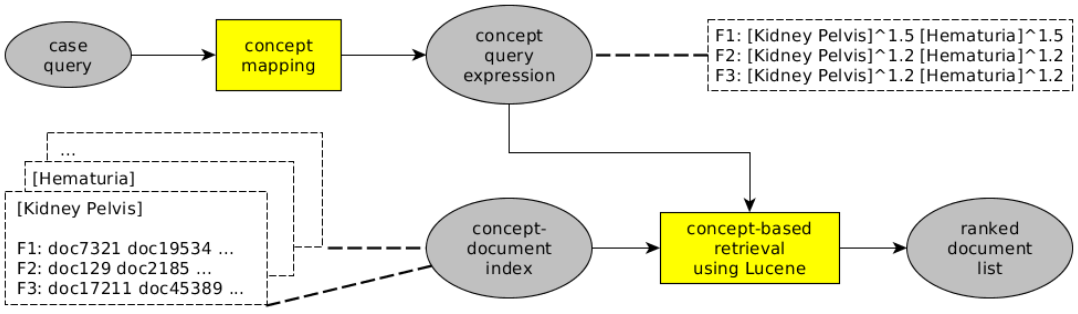


Figure 6.1: Concept-based retrieval process showing examples for query expression and concept-document index.

in the figure). In Fig. 6.1 the index fields F1, F2, and F3 correspond to the three MeSH annotation types described earlier.

To prepare a case query for concept-based retrieval, it needs to be transformed to a concept vector or, more specifically in the employed Lucene framework, a query expression specifying MeSH node identifiers and boosting factors for the three index fields to search in. MeSH concepts supposed to be relevant for the given query can be obtained by any concept mapping method described in Chapter 4. For every MeSH node identifier corresponding to such concepts, three Lucene query terms are generated, each addressing a different index field with corresponding boosting factor, as described above. Executing the resulting query expression using Lucene's retrieval engine and the previously built concept-document index then retrieves a ranked list of indexed documents according to the paradigm of concept-based retrieval.

6.2 Experiments

As explained in the introduction to this chapter, the main purpose of experiments is to evaluate different concept mapping algorithms introduced in Chapter 4 by measuring their effectiveness for concept-based retrieval on the MCR dataset. All experiments use the same concept-based document index with separate fields for manually annotated and automatically annotated MeSH terms represented as node identifiers, as described in Section 6.1. The different concept mapping algorithms are applied to case queries to transform them to query expressions containing MeSH node identifiers only. Query expressions are then executed using the Lucene retrieval engine, resulting in a ranked list of 100 documents[1]. The result list is evaluated using common IR performance measures like *precision at 10* and *mean average precision* (MAP), as described in Section 2.2.2.

[1]Restriction of the result list to 100 top-ranked documents follows the official evaluation procedure of the ImageCLEF MCR challenge [104].

Experiments were conducted for text-to-concept mapping algorithms (described in Section 6.2.1) and for image-to-concept mapping techniques (Section 6.2.2) that were already described in Chapter 4. The effectiveness of these algorithms for concept-based retrieval is additionally compared to plain fulltext retrieval and to concept-based retrieval based on an "ideal" concept mapping algorithm that obtains MeSH concepts from documents that are actually relevant to a given case query.

6.2.1 Text-to-Concept Mapping Algorithms

Text-to-concept mapping algorithms used in experiments are almost the same as those evaluated in Chapter 4: MetaMap, OBA, and the three string matching algorithms BinCov, Dist, and BinDist. Instead of the nearest-neighbor classifier MeshUp, whose web service was no longer available when concept-based retrieval experiments started, we used our own implementation of a kNN classifier which uses the text query for fulltext retrieval and ranks MeSH annotations of the top k retrieved documents according to an accumulated score derived from documents they appear in (see Section 4.2.2). Results presented in this section were obtained in cooperation with Florian Winkler [226].

Parameters of text-to-concept mapping algorithms include the ones described in Chapter 4, namely score thresholds for concept selection and a specialty boosting flag, but concept-based retrieval and the kNN classifier add additional parameters. In more detail, the parameters of text-to-concept mapping algorithms used in experiments are:

minscore Score threshold used to select concepts for MetaMap, OBA, and string matching algorithms.

specialtyboost Flag indicating whether to use specialty boosting for string matching algorithms (see Section 4.2.3.7 on page 75).

maxrank Rank threshold used to select concepts for kNN classifier.

knn_k Number of nearest-neighbor documents used by kNN classifier.

fieldboost Boosting factors $(\alpha_1, \alpha_2, \alpha_3)$ used for different MeSH annotation types to construct the query expression (see Section 6.1). The kNN classifier uses the same boosting factors to compute concept scores by Equation (4.1).

Thorough parameter optimization would require cross-validation on the query set—given the 35 queries of the MCR dataset, a 5-fold cross-validation would be suitable. However, since our evaluation aims at comparing concept mapping algorithms and does not focus on measuring "true" concept-based retrieval performance, we decided to relinquish cross-validation and to optimize parameters on the entire query set. Measured

Table 6.1: Parameters of text-to-concept mapping algorithms optimized for concept-based retrieval. Maximal values of the searched range indicated by * were chosen adaptively.

Algorithm	*minscore*	*specialtyboost*	*maxrank*	*knn_k*	*fieldboost*
Range	0–max*	yes, no	1–max*	1–20	0–2.5
MetaMap	12.94	–	–	–	1.8, 1.4, 1.0
OBA	32759	–	–	–	2.0, 1.6, 1.0
kNN	–	–	33	6	2.2, 1.5, 1.0
BinCov	3.0	yes	–	–	1.0, 1.0, 1.0
Dist	2.6	yes	–	–	2.0, 1.5, 1.0
BinDist	0.74	yes	–	–	1.1, 1.0, 1.0

retrieval performance can therefore be regarded as an upper bound of "true" effectiveness.

Optimized parameters for text-to-concept mapping algorithms used for concept-based retrieval experiments are presented in Table 6.1. The range used to search for optimal values is displayed near the top of the table. In analogy to experiments described in Chapter 4, maximal values of *minscore* were chosen algorithm-specific, and *maxrank* was increased gradually until retrieval performance started to drop. Since evaluation of a single parameter configuration requires a concept-based retrieval run on all queries, a greedy coarse-to-fine approach was applied to search for optimal values: first, parameters were evaluated on a coarse grid of values covering the search range; then a fine grid in the region of best performing parameter combinations was used to determine optimal values.

Experimental results showing concept-based retrieval performance on the MCR dataset when text-to-concept mapping algorithms are applied to case queries are presented in Table 6.2 and Fig. 6.2. To enable judgment of these algorithms beyond mutual comparison, performance numbers for fulltext retrieval and an "ideal" concept mapping algorithm *MGT* (MeSH ground truth) are provided as well. The MGT algorithm uses manual MeSH annotations of actually relevant documents (using ground-truth relevance judgments) for concept mapping of a given case query. Concept-based retrieval performance of the MGT algorithm therefore represents a tight upper bound for any concept mapping algorithm and reveals the potential effectiveness of concept-based retrieval methods.

Results of Table 6.2 confirm that concept-based retrieval using MeSH concepts is less effective than fulltext retrieval, although concept-based retrieval using the kNN classifier performs almost as well as fulltext retrieval with respect to MAP. The relatively good performance of the kNN classifier can be explained by its use of fulltext

Table 6.2: Concept-based retrieval performance of text-to-concept mapping algorithms. Percentages denote changes relative to fulltext retrieval.

Algorithm	P@10		MAP	
Fulltext	0.234		0.160	
MGT	0.480	+105%	0.369	+131%
MetaMap	0.066	-72%	0.042	-74%
OBA	0.097	-59%	0.056	-65%
kNN	0.206	-12%	0.156	-3%
BinCov	0.009	-96%	0.003	-98%
Dist	0.026	-89%	0.011	-93%
BinDist	0.100	-57%	0.060	-63%

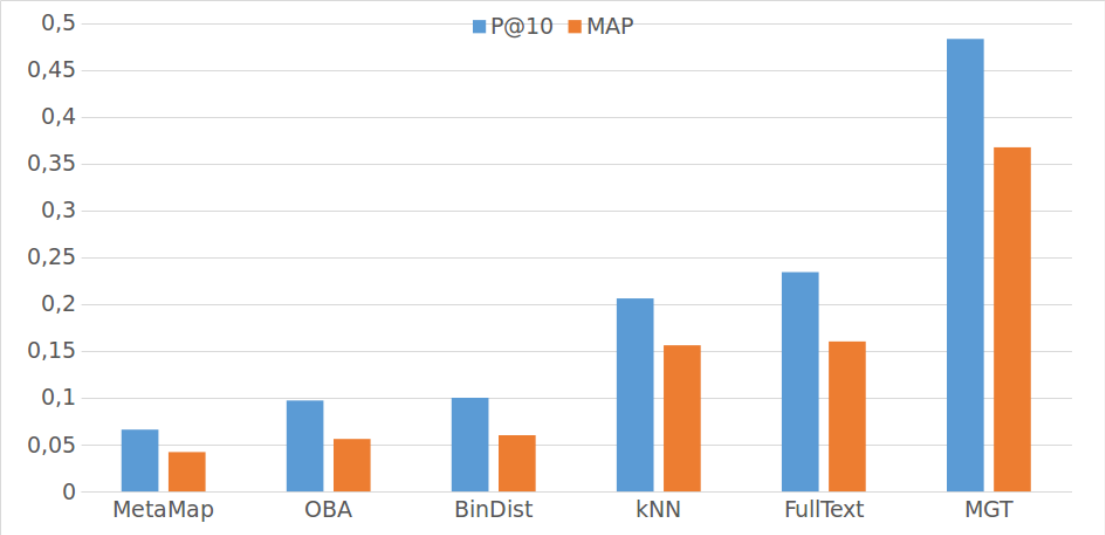


Figure 6.2: Concept-based retrieval performance of text-to-concept mapping algorithms.

retrieval to obtain documents for harvesting MeSH concepts, leading to a similar effectiveness as fulltext retrieval. Among other text-to-concept mapping algorithms, the BinDist string matching algorithm was most effective and lead to a slightly better retrieval performance than OBA and MetaMap. The high retrieval performance of the MGT algorithm indicates that concept-based retrieval can potentially be more effective than fulltext retrieval and motivates future research in the field of concept mapping algorithms.

6.2.2 Image-to-Concept Mapping Algorithms

Experiments evaluate the effectiveness of image-to-concept mapping algorithms described in Section 4.3 for concept-based retrieval. All proposed image-to-concept mapping algorithms are nearest-neighbor (kNN) classifiers using a Lucene index of images found in biomedical articles of the MCR dataset. MeSH concepts of indexed images were automatically determined by applying the BinDist string matching algorithm to image captions, using optimized parameters determined by text-to-concept mapping experiments of Chapter 4. The image index can be searched by both concepts and content-based visual descriptors (CEDD, FCTH, and PHOG, see Section 4.3).

We consider three variants of kNN-based image-to-concept mapping algorithms for experimental evaluation. They differ in the way how the image index is used to retrieve images that are potentially relevant to a given case query.

M1 For each image contained in the case query, CEDD and FCTH features are extracted, and each feature is used to retrieve k images from the image index. If the case query contains n images, this method generates $2n$ image lists of length k for harvesting MeSH concepts.

M2 The case query is mapped to MeSH concepts using the best-performing text-to-concept mapping algorithm according to Section 6.2.1 (kNN classifier), and the result is used to perform concept-based retrieval on the image index, generating a single image list of length k. Note that the resulting kNN classifier does not utilize content-based visual information, but as the image index is involved, we categorize this approach as image-to-concept mapping.

M3 Method M2 is applied to retrieve 100 images, which are then reranked according to visual similarity with each image of the case query, resulting in n image lists. Each list is then truncated to length k. Visual similarity is determined using CEDD features only.

Although methods M2 and M3 may be too inefficient for practical purposes (M2 involves three retrieval operations until the final concept-based retrieval result is obtained), they are proposed and evaluated in an attempt to reduce the semantic gap between case query and retrieved images which is inherent to method M1. For M2 and M3, biomedical concepts relevant to the case query are expected to improve the semantic relationship with retrieved images.

MeSH concepts are harvested from image lists generated by one of these methods, and concept scores are accumulated over all image lists according to Equations (4.1) and (4.11), as explained in Chapter 4. Finally, concept-based retrieval is performed by applying the same query formulation method and document index as used for evaluating text-to-concept mapping algorithms (see Section 6.2.1).

Table 6.3: Parameters of image-to-concept mapping methods used for experiments, and average number of concepts obtained per case query.

Method	*knn_k*	*fieldboost*	*concepts per query*
M1	3	1.5, 1.2, 1.0	40.3
M2	3	1.5, 1.2, 1.0	46.3
M3	3	1.5, 1.2, 1.0	37.0

Table 6.4: Concept-based retrieval performance of image-to-concept mapping algorithms. Percentages denote the performance ratio with respect to fulltext retrieval.

Algorithm	*P@10*		*MAP*	
Fulltext	0.234	100%	0.160	100%
M1	0.015	6%	0.003	2%
M2	0.111	48%	0.049	30%
M3	0.085	36%	0.039	24%

The tested image-to-concept mapping methods support only two parameters already known from text-to-concept mapping experiments: *knn_k* is the number k of images to retrieve as described above; and *fieldboost* specifies the reliability factors of MeSH annotation types as described in Section 6.2.1. The *maxrank* parameter limiting the number of harvested MeSH concepts has not been implemented, because the number of concepts is implicitly limited by *knn_k* and time constraints prohibited a comprehensive implementation of this experiment. For the latter reason, parameter optimization could not be performed systematically, but parameters were chosen manually based on the experience of a few experimental runs. Table 6.3 documents parameter settings used for evaluation of image-to-concept mapping experiments and the resulting average number of concepts obtained per case query.

Table 6.4 and Fig. 6.3 display experimental results obtained for concept-based retrieval after applying image-to-concept mapping algorithms M1, M2, or M3 to case queries of the MCR dataset. As expected, concept mapping using visual similarity (M1) results in very poor performance compared to the fulltext retrieval baseline, indicating that visual similarity induced by global image features is rarely able to find images that are relevant for given query images.

M2 shows the best concept-based retrieval performance of all three evaluated algorithms, but still displays a much lower performance than fulltext retrieval. Since we know from Section 6.2.1 that the kNN classifier used to map query text to MeSH concepts performs well, the loss in retrieval performance for M2 is caused by a mismatch of query concepts and concept annotations in the image index. The mismatch may have two reasons: first, not all concepts mapped from queries may occur in the image index;

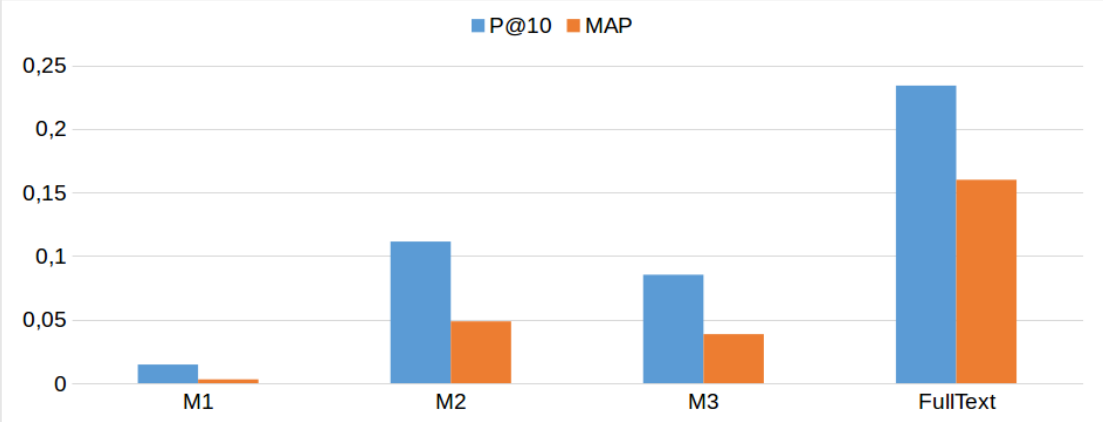


Figure 6.3: Concept-based retrieval performance of image-to-concept mapping algorithms.

and second, there are no or too few images annotated with multiple concepts that are relevant to the case query, so that only partially relevant images are retrieved. Since both conditions may be caused by the method of obtaining MeSH annotations of indexed images—a string matching algorithm is applied to image captions—, we conclude that a more powerful method of MeSH annotation for images of medical case descriptions is required that is able to map a given image to several relevant MeSH concepts. Note that such a powerful MeSH annotation method could also improve method M1, because chances are higher that more relevant MeSH concepts can be harvested from visually similar neighbors of query images.

Visual reranking applied by method M3 after concept-based retrieval on the image index turned out to deteriorate results compared to method M2. This can be explained by the rare semantic relatedness of visually similar images already observed for method M1, resulting in a more or less "random" permutation of images returned by concept-based retrieval according to method M2, which causes scores of relevant concepts harvested from these image lists to decrease on average.

6.3 Summary

This chapter investigated the effectiveness of several concept mapping algorithms described in Chapter 4 for concept-based retrieval, which uses concept vector representations of both documents and queries for relevance ranking. Experiments used the Lucene text retrieval engine for concept-based indexing and retrieval, representing MeSH concepts by their MeSH node identifiers. All experiments used the same concept-based index of articles of the MCR dataset, where MeSH concepts were obtained from manual

MeSH annotations as well as from automatic ones produced by the BinDist text-to-concept mapping algorithm. The different concept mapping algorithms were applied to represent case queries by MeSH concepts.

Evaluation of concept mapping algorithms confirmed results of similar studies found in the literature, namely that (1) concept-based retrieval using practical concept mapping algorithms does not improve over plain fulltext retrieval, and (2) nearest-neighbor classifiers represent the most effective concept mapping algorithms for concept-based retrieval. The first result can be explained by the difficulty of finding *multiple* relevant concepts for a given case query, because an "ideal" concept mapping algorithm delivering concepts of actually relevant documents displayed a concept-based retrieval performance that outperformed fulltext retrieval by more than 100%. Regarding the second result, our kNN classifier implementing text-to-concept mapping achieved a slightly lower concept-based retrieval performance than fulltext retrieval, which is also explained by the good performance of the "ideal" concept mapping algorithm: as concept-based retrieval is effective when relevant query concepts can be found, the effectiveness of the kNN classifier is governed by its ability to find documents with relevant concepts; and since the kNN classifier uses fulltext retrieval for this purpose, its overall concept-based retrieval performance is similar to fulltext retrieval.

When comparing evaluation results for different text-to-concept mapping algorithms, similar conclusions as for text classification experiments in Chapter 4 can be drawn: the kNN classifier outperforms all other concept mapping algorithms by large margins, the BinDist algorithm is the most effective from the set of tested string matching approaches, and the existing concept mapping systems MetaMap and OBA are not more effective than BinDist.

Tested image-to-concept mapping algorithms were instances of kNN classifiers utilizing an image index to retrieve images that are potentially relevant to a given case query. The achieved concept-based retrieval performance was low compared to fulltext retrieval, with MAP ratios ranging from only 2% for a kNN classifier based on visual similarity (using global image features) to 30% for a kNN classifier performing concept-based retrieval on the image index. The main reason for the low effectiveness of tested image-to-concept mapping algorithms is seen in the difficulty of associating *multiple* relevant concepts with article images of the MCR dataset, required to build the image index. Compared to text-to-concept mapping algorithms, tested image-to-concept mapping approaches were not more effective than BinDist.

CHAPTER 7

Multimodal Retrieval

In this chapter, we investigate methods that utilize multiple representations of case queries and case descriptions for medical case retrieval (MCR) with the ultimate goal of improving retrieval performance over unimodal retrieval. *Modalities* of multimedia documents are representations of information from certain sources or with a certain added value. For case descriptions, we consider textual, visual, and concept-based modalities, where textual and visual modalities refer to different information sources (article text and images) and the concept-based modality represents information obtained either from manual annotations or from textual or visual sources, but with added value from a controlled biomedical vocabulary.

Section 7.1 proposes a framework for multimodal retrieval that combines the most effective MCR methods identified in previous chapters, namely textual retrieval with query expansion and concept-based retrieval. Since concept-based representations may be derived from textual or visual information (or both), the proposed framework is capable of utilizing all modalities available with case descriptions and queries of the MCR dataset.

The method combining textual and concept-based retrieval according to the proposed framework is known as *late fusion*, because both retrieval methods are applied separately to produce ranked document lists, which are then fused into a single result list. Section 7.2 considers two different late fusion methods: *linear fusion* (Section 7.2.1) combines scores or ranks of a document retrieved by both retrieval methods by a linear expression whose weights are fixed for the multimodal retrieval system; *query-adaptive fusion* (Section 7.2.2), on the other hand, chooses linear fusion weights for each query separately, based on predictions of retrieval performance of both component systems for the given query. To estimate the potential of query-adaptive fusion (QAF) for multimodal retrieval, we use an "ideal" QAF variant for experiments, where predicted query performance is replaced by the actual retrieval performance of component systems measured using ground-truth information.

Experimental results are presented and discussed in Section 7.3, and Section 7.4 summarizes results and findings of this chapter.

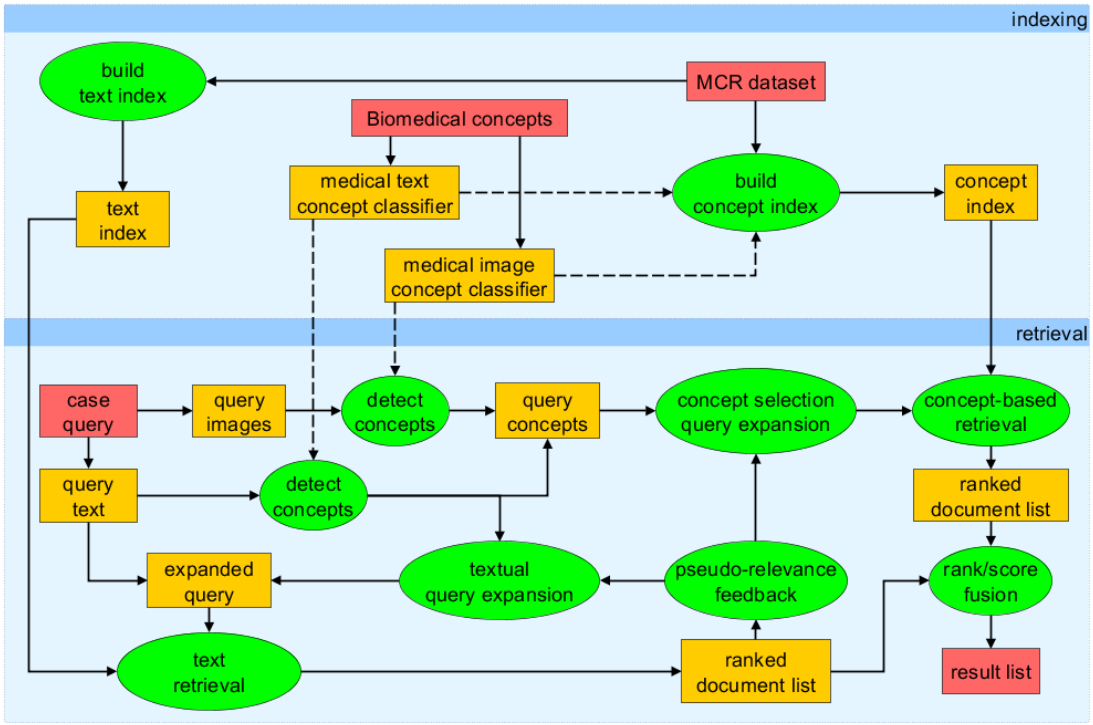


Figure 7.1: Proposed multimodal retrieval framework for MCR.

7.1 Proposed Framework

By combining textual retrieval including query expansion and concept-based retrieval using late fusion, we obtain a framework for multimodal MCR that integrates most of the techniques covered in previous chapters of this thesis. A flow chart of the proposed retrieval framework is presented in Fig. 7.1, representing both offline indexing of case descriptions and online retrieval for a given case query.

During indexing, two separate inverted indexes of medical case descriptions (documents) are generated: the *text index* represents a conventional fulltext index searchable by indexed terms (stemmed words of documents after stop word removal); the *concept index* is built from the same documents by mapping them to biomedical concepts (of a controlled vocabulary) using multi-label concept classifiers applied to text or images (or both) of case descriptions. The concept index is searchable by biomedical concepts represented by unique identifiers.

Retrieval processing for a given case query proceeds in two parallel data paths that may be interconnected. One path executes *fulltext retrieval* using the textual modality of the query and the text index. Text retrieval may be enhanced by query expansion methods utilizing two different data sources: biomedical concepts derived from the

query and document terms obtained from pseudo-relevance feedback, as described in Chapter 5.

The second retrieval path performs *concept-based retrieval* by applying concept mapping algorithms (see Chapter 4) to one or both modalities of the case query. Note that the framework also accommodates concept mapping by multi-view learning, which uses both query modalities to predict query concepts, although Fig. 7.1 depicts unimodal concept mapping only. Candidate query concepts then undergo concept selection (e.g. by applying a rank threshold to ranked concept lists) and optionally query expansion by textual pseudo-relevance feedback—this kind of query expansion has not been implemented in our experiments, however, because the most effective concept mapping algorithms, namely nearest-neighbor classifiers, already obtain concepts from pseudo-relevant documents. The resulting concept vector representation of the query is finally used to retrieve documents from the concept index, as described in Chapter 6.

Fulltext and concept-based retrieval processes produce separate ranked document lists that are combined into a final result list by one of the fusion methods described in subsequent Section 7.2. This late fusion approach may improve precision or recall (or both) of multimodal retrieval, depending on whether both retrieval processes rank the same relevant documents high, or retrieve different relevant documents.

7.2 Fusion Methods

The *late fusion* problem considered by the information (or data) fusion literature [215] can be stated as follows: given two or more information retrieval (IR) systems (called *component* systems) applied to the same document collection and query set, find a method for combining (*fusing*) the resulting ranked lists of retrieved documents into a single ranked list such that retrieval performance of the fused list improves over each component system. For practical purposes, the parameters of the fusion method must be determined without ground-truth knowledge (relevance judgments of retrieved documents) about the actual retrieval performance of component systems. For research purposes, however, "ideal" parameters obtained from ground-truth knowledge may serve to evaluate the potential effectiveness of fusion methods.

From the various late fusion methods known from information and multimedia retrieval literature [59, 89, 106, 122, 6, 158, 229, 249], we selected two approaches that proved to be effective on general multimodal datasets: *linear fusion* [229] is a popular method combining scores or ranks of documents retrieved by component systems by linear expressions with fixed weights; *query-adaptive fusion* [106] develops linear fusion further by making linear combination weights dependent on a given query, which usually requires to predict the retrieval performance of component systems for each query presented to the system. Details of these two fusion methods, which were applied to our

proposed multimodal retrieval framework in experiments, are described in Sections 7.2.1 and 7.2.2, respectively.

7.2.1 Linear Fusion

The linear fusion method proposed by Wu [229] comprises two regression techniques that aim to infer optimal fusion weights for component systems from training data (ranked lists of documents retrieved by each component system for a given set of training queries, and corresponding relevance judgments). First, *logistic score normalization* learns reliable scores from ranks of retrieved documents from training data by logistic regression. Then *multiple linear regression* is applied to normalized scores of retrieved documents in the training set to learn fusion weights of component systems.

Logistic score normalization models the probability of relevance, $s_A(r)$, of a document retrieved at rank position r ($r \geq 1$) by component system A by a logistic function of the logarithm $\ln(r)$:

$$s_A(r) = \frac{1}{1 + e^{-a-b*\ln(r)}} \tag{7.1}$$

where a and b are real numbers dependent on A that are determined by fitting the logistic function to binary relevance values of retrieved documents in a training set (binary logistic regression). $s_A(r)$ is then used as normalized relevance score ($0 \leq s_A(r) \leq 1$) of a document retrieved by A at rank r. Wu [229] provides empirical evidence that other score normalization models—including linear normalization that directly normalizes original retrieval scores of component system A—do not provide better estimates of the actual binary relevance of retrieved documents than logistic score normalization.

In light of the proposed multimodal retrieval framework, we restrict the following presentation to two component systems, although Wu's method is defined for any number of component systems. Let D_q be the union of document sets retrieved by two component systems A and B for a given query q, and let $s_A(q,d)$ and $s_B(q,d)$ be the scores of retrieved document $d \in D_q$ obtained by logistic score normalization. If a component system, say B, did not retrieve d, then set $s_B(q,d) = 0$. The score of document d after linear fusion is then defined as:

$$S_\beta(q,d) = \beta * s_A(q,d) + (1-\beta) * s_B(q,d) \tag{7.2}$$

where β and $1-\beta$ are fusion weights assigned to component systems A and B, respectively. Note that $0 \leq S_\beta(q,d) \leq 1$, because $s_A(q,d)$ and $s_B(q,d)$ are normalized scores. The problem of linear fusion of two component systems therefore reduces to determine an optimal value for β, which we denote as $\hat{\beta}$.

In general, fusion weights of multiple component systems can be determined by multiple linear regression that fits fused scores to actual binary relevance scores of retrieved documents in a training set. For two component systems, $\hat{\beta}$ can be determined by simple linear regression that solves the following least squares problem:

$$\hat{\beta} = \underset{0 \leq \beta \leq 1}{\operatorname{argmin}} \sum_{q \in Q} \sum_{d \in D_q} \left(S_\beta(q,d) - y(q,d) \right)^2 \tag{7.3}$$

where Q is the set of training queries, and $y(q,d)$ is the ground-truth binary relevance score of document $d \in D_q$ retrieved for query q.

We claim that solving the regression problem (7.3) is equivalent to optimizing parameter β with respect to mean average precision (MAP) of the fused retrieval system on the training set. To recognize the claim, suppose that β' is a fusion weight leading to a sub-optimal least squares expression (7.3). Then there is a query q and a document $d \in D_q$ such that

$$\left(S_{\hat{\beta}}(q,d) - y(q,d) \right)^2 < \left(S_{\beta'}(q,d) - y(q,d) \right)^2 . \tag{7.4}$$

If $y(q,d) = 1$, i.e. document d is relevant for query q, it follows that $0 \leq S_{\beta'}(q,d) < S_{\hat{\beta}}(q,d) \leq 1$. Since the fused result lists for query q, denoted as $R_{\hat{\beta}}(q)$ and $R_{\beta'}(q)$, are ranked according to $S_{\hat{\beta}}$ and $S_{\beta'}$, respectively, the relevant document d is ranked lower in $R_{\beta'}(q)$ than in $R_{\hat{\beta}}(q)$. According to the definition of MAP (see Section 2.2.2), this means that $\mathrm{MAP}(R_{\beta'}(q)) < \mathrm{MAP}(R_{\hat{\beta}}(q))$. Analogously, if $y(q,d) = 0$, i.e. document d is not relevant for query q, then d is ranked higher in $R_{\beta'}(q)$ than in $R_{\hat{\beta}}(q)$, resulting in the same conclusion that $\mathrm{MAP}(R_{\beta'}(q)) < \mathrm{MAP}(R_{\hat{\beta}}(q))$. Hence, a solution $\hat{\beta}$ of the least squares problem (7.3) leads to a maximal MAP value. A similar line of argumentation shows that also the converse implication is true, confirming the claim.

7.2.2 Ideal Query-Adaptive Fusion

Query-adaptive fusion (QAF) is an extension of linear fusion that determines query-specific fusion weights based on a prediction of retrieval performance of component systems for a given query. Although algorithms for estimating query-specific retrieval performance from ranked document lists are available [56, 55, 187, 252], we restricted experiments to "ideal" query-adaptive fusion, where query-specific retrieval performance of component systems is calculated using ground-truth relevance judgments. Experimental results should therefore provide an upper bound of the effectiveness that can be achieved by practical QAF systems, which could be the subject of further work.

Given performance predictions p_A and p_B of component systems A and B for a certain query q, respectively, we use the performance square weighting scheme [230] to derive a query-specific fusion weight β_q that takes the role of β in Equation (7.2):

$$\beta_q = \frac{p_A^2}{p_A^2 + p_B^2} \tag{7.5}$$

Note that β_q is the fusion weight of component system A, and $1 - \beta_q$ is the weight of component system B. Result lists of component systems for query q are then fused in the same manner as described in Section 7.2.1. For experiments with "ideal" QAF, p_A and p_B were taken as the average precision (see Section 2.2.2) achieved by component systems for query q.

Performance square weighting was chosen, because it proved to be superior over a number of other proposed non-adaptive fusion methods [230, 229] and can easily be applied to query-adaptive fusion (whereas the linear regression method described in Section 7.2.1 cannot).

7.3 Experiments

Experiments evaluate the effectiveness of the proposed multimodal retrieval framework (Section 7.1) by instantiating the framework with best-performing text-based and concept-based retrieval systems determined in Chapters 5 and 6, respectively. As in Chapter 6, experiments include an "ideal" concept-based retrieval system to estimate an upper bound for resulting retrieval performance.

The focus of experiments is laid on assessing and analyzing the effectiveness of fusion methods described in Section 7.2. Results of linear fusion experiments are presented in Section 7.3.1, query-adaptive fusion experiments are the subject of Section 7.3.2. Fusion experiments used the following component retrieval methods found to be most effective in previous Chapters:

T Best text-based retrieval method (Mt4x1r2+2) determined by ImageCLEF evaluation method in Section 5.3.5 (see Table 5.9 on page 115). The method employs MeSH query expansion followed by pseudo-relevance feedback with unigrams and bigrams, and document expansion with MeSH terms.

C Best practical concept-based retrieval method (kNN) identified in Section 6.2.1 (see Table 6.2 on page 123), which obtains MeSH concepts from nearest-neighbor documents retrieved by fulltext search.

C+ "Ideal" concept-based retrieval method (MGT, see Table 6.2), which obtains MeSH concepts from actually relevant documents, as determined by ground-truth relevance judgments.

Table 7.1: Training samples and learned model parameters for logistic score normalization of component retrieval methods.

Method	*Training samples*			*Model parameters*	
	relevant	non-relevant	total	a	b
T	370	1895	2265	-0.1716	-0.4402
C	290	1086	1376	-0.2400	-0.3362
C+	261	291	552	3.1819	-1.1781

Method **T**

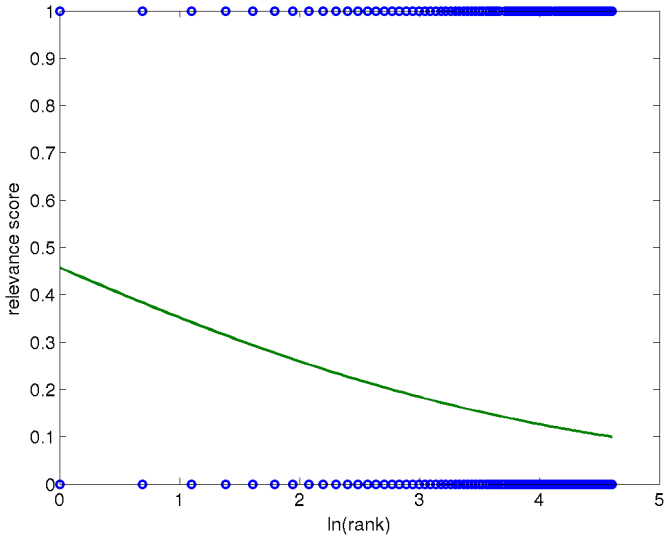
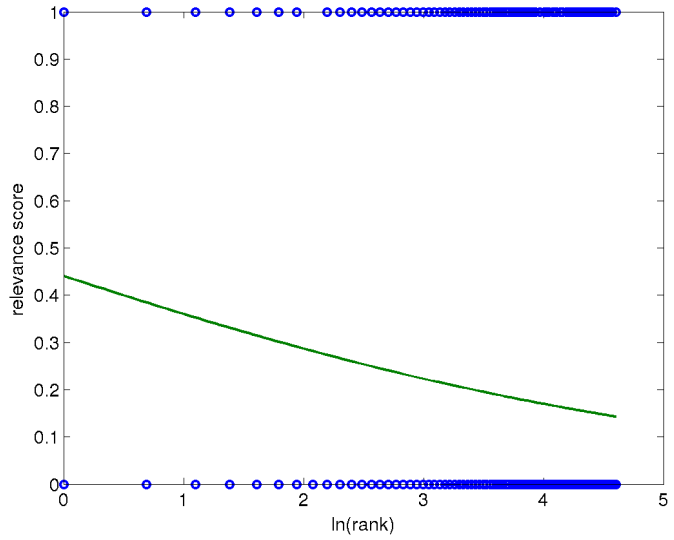


Figure 7.2: Training samples (plotted as circles) and learned logistic curve for score normalization of text-based component retrieval method.

Logistic score normalization was performed for each component retrieval method on the MCR dataset, using all 35 queries and ground-truth relevance judgments for training. Table 7.1 presents the number of available training samples and learned logistic model parameters a and b (see Equation (7.1)). Training samples are judged documents retrieved by a given component retrieval method, hence the number of available training samples depends on the retrieval method. Plots of training samples and learned logistic curves for each retrieval method are shown in Figures 7.2 and 7.3. Training samples appear as circles at relevance score 0 (not relevant) or 1 (relevant). The S-shape of the logistic curve is most pronounced for retrieval method C+, indicating a stronger correlation between relevant training samples and low ranks (and between non-relevant samples and high ranks) than for other retrieval methods. According to learned logistic curves, normalized scores for method C+ cover almost the full range of relevance scores between 0 and 1, whereas normalized scores for methods T and C never exceed 0.5.

Method **C**



Method **C+**

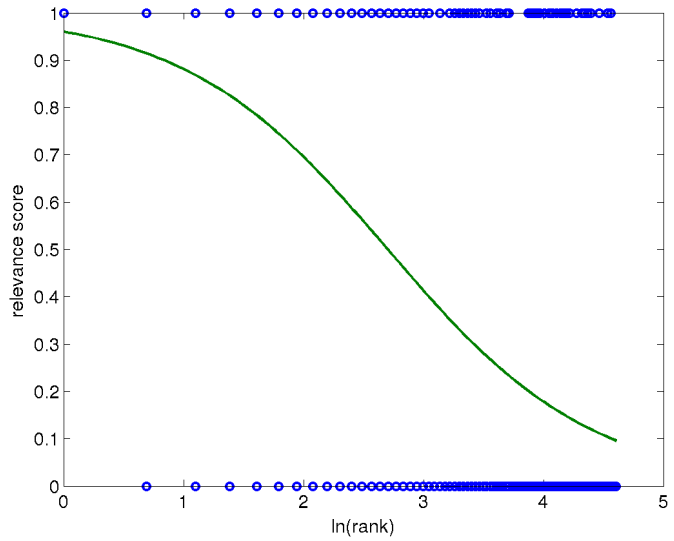


Figure 7.3: Training samples (plotted as circles) and learned logistic curves for score normalization of concept-based component retrieval methods.

Using all available queries and training samples leads to most effective score normalization with respect to subsequent retrieval performance of the fused system. We also tried to use only the best performing queries for each component system for training, and applied a rank threshold to training samples to select only low-ranked relevant documents and high-ranked non-relevant ones. The resulting logistic models had a more pronounced S-shape and covered a greater range of relevance scores, but retrieval performance of the fused system dropped. Hence using representative training samples for logistic score normalization is important.

7.3.1 Linear Fusion Results

For linear fusion experiments, the optimal value of fusion weight β could be determined by parameter optimization with respect to mean average precision (MAP) of the fused result list, as explained in Section 7.2.1. Following the evaluation methodology of Chapter 6, we did not apply cross-validation to decouple query sets used for parameter optimization and testing, because we are mainly interested in upper bounds for the effectiveness of fusion methods. Parameter optimization was therefore performed on the entire query set, which was also used for testing. Note that this simplified evaluation procedure is consistent with the applied training method for logistic score normalization, which also used the entire query set.

Table 7.2 presents optimized fusion weights and resulting retrieval performance on the MCR dataset when combining text-based component system T and concept-based component system C or C+ by linear fusion after logistic score normalization. Both fusion runs achieved higher performance values than their component systems, meaning that fusion helped to rank relevant case descriptions higher on average. For fusion of T and C, the optimal fusion weight $\beta = 0.87$ indicates that document scores assigned by method T were weighted higher than scores of method C, which seems reasonable given the better retrieval performance of component system T compared to C. However, the optimal weight for fusing T and C+ ($\beta = 0.67$) also favors the text-based component system T, although the ideal concept-based component system C+ displays substantially better performance numbers than T. This apparent inconsistency can be explained by the different score ranges covered by logistic normalization of scores produced by systems T and C+ (see Figures 7.2 and 7.3). The higher fusion weight of component system T is overcompensated by smaller values resulting from score normalization compared to component system C+.

The dependency of linear fusion performance on weight β of the text-based component system is presented in Fig. 7.4, showing achieved mean average precision of both fusion runs for 11 equally spaced values of β. Note that parameter optimization used a finer grid of step size 0.01 to determine the optimal fusion weight β. Whereas fusion of practical component systems T and C improves only slightly over T for a narrow range

Table 7.2: Retrieval performance of linear fusion (L) of text-based (T) and concept-based (C, C+) retrieval with optimized fusion weight β on MCR dataset. Percentages denote the performance ratio with respect to method T.

Method	β	*P@10*		*MAP*	
T	1.00	0.257	100%	0.245	100%
C	0.00	0.206	80%	0.156	64%
L(T,C)	0.87	0.269	105%	0.252	103%
C+	0.00	0.480	187%	0.369	151%
L(T,C+)	0.67	0.491	191%	0.440	180%

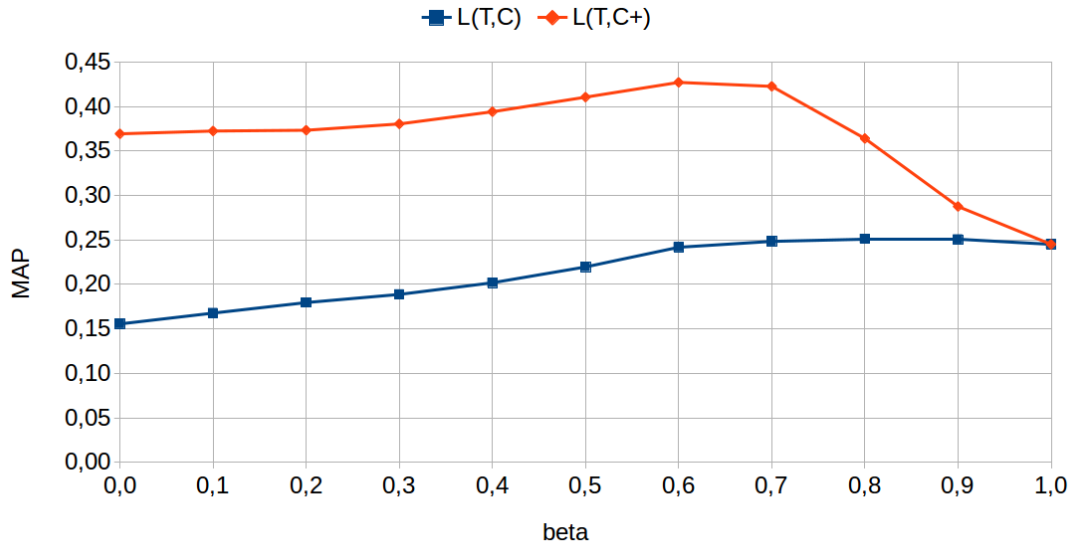


Figure 7.4: Retrieval performance of linear fusion (L) of text-based (T) and concept-based (C, C+) retrieval for different values of fusion weight β.

of fusion weights, fusion of T and the ideal concept-based system C is more effective than any component system for a broad range of fusion weights. This can again be explained by the different score ranges resulting from logistic score normalization of component systems T and C+. Starting from $\beta = 0$, increasing fusion weights start adding relevant documents retrieved by T to the fused list, but still keep relevant documents retrieved by C+ at low ranks due to their high normalized score. Only for $\beta > 0.7$ ranks of relevant documents retrieved by C+ may start to increase, leading to lower MAP than achieved by C+. On the other hand, normalized score ranges for T and C are similar, resulting in a more pronounced "mixing" of component ranking lists that often deteriorates ranks of relevant documents retrieved by T, resulting in decreased MAP of fusion for most values of β compared to T.

Table 7.3: Retrieval performance of ideal query-adaptive fusion (Q) of text-based (T) and concept-based (C, C+) retrieval on MCR dataset. Percentages denote the performance ratio with respect to method T.

Method	*P@10*		*MAP*	
T	0.257	100%	0.245	100%
C	0.206	80%	0.156	64%
Q(T,C)	0.314	122%	0.267	109%
C+	0.480	187%	0.369	151%
Q(T,C+)	0.520	202%	0.449	183%

7.3.2 Query-Adaptive Fusion Results

Experiments for query-adaptive fusion used performance square weighting by means of an oracle that provided the actual retrieval performance (MAP) of component systems for a given query, as described in Section 7.2.2. Results for ideal query-adaptive fusion of text-based (T) and concept-based (C, C+) component systems measured on the MCR dataset are presented in Table 7.3.

Results show similar characteristics as linear fusion results (Section 7.3.1): both fusion runs improved retrieval performance over each component system. However, query-adaptive fusion consistently displays higher performance gains than linear fusion, with a more pronounced improvement in early precision (P@10) than in MAP.

To investigate the reasons for the increase in retrieval performance achieved by ideal query-adaptive fusion, we performed a per-query recall analysis of retrieved ranking lists. We are interested in two aspects of achieved recall: (1) the ratio of the number of relevant documents retrieved by fusion and the corresponding number achieved by both component systems, which we define as *recall efficiency* of fusion; and (2) the number of relevant documents retrieved by fusion that were exclusively retrieved by concept-based retrieval, from which we derive a ratio denoted as *C-utility* (utility of concept-based retrieval for fusion).

More precisely, recall analysis was conducted by calculating the following numbers from ranking lists produced by fusion and component systems for a given query: fusion recall ρ_F, maximal fusion recall ρ_F^*, recall efficiency φ, C-utility ω_C, and maximal C-utility ω_C^*. To define these measures formally, let R_T, R_C, and R_F be the sets of relevant documents retrieved by text-based retrieval, concept-based retrieval, and fusion, respectively, and let R be the set of all (judged) relevant documents for the given query. We then define the following recall-based measures:

$$\rho_F = \frac{|R_F|}{|R|} \tag{7.6}$$

$$\rho_F^* = \frac{|R_T \cup R_C|}{|R|} \tag{7.7}$$

$$\varphi = \frac{\rho_F}{\rho_F^*} = \frac{|R_F|}{|R_T \cup R_C|} \tag{7.8}$$

$$\omega_C = \frac{|R_F \setminus R_T|}{|R_T \cup R_C|} \tag{7.9}$$

$$\omega_C^* = \frac{|R_C \setminus R_T|}{|R_T \cup R_C|} \tag{7.10}$$

In cases where the denominator of a ratio is zero, we define the corresponding measure as zero. Note that values of all defined measures are constrained to the range $[0, 1]$, and that $\rho_F \leq \rho_F^*$ and $\omega_C \leq \omega_C^*$ hold. Measures ρ_F^* and ω_C^* are called *maximal*, because their definitions employ the largest sets of relevant documents ($R_T \cup R_C$ and $R_C \setminus R_T$, respectively) a fusion algorithm can produce, given the two component systems.

Results of per-query recall analysis for ideal query-adaptive fusion of T and C+ component systems are presented in Table 7.4. We chose to analyze Q(T,C+) rather than Q(T,C), because both fusion systems were constructed as "ideal" systems with respect to parameter optimization and query performance prediction, but Q(T,C+) provided better retrieval performance.

For 10 out of 35 queries, recall efficiency is 1, meaning that QAF retrieved all relevant documents that were retrieved by any component system. The smallest recall efficiency was achieved for query 1 (0.38), where component system T retrieved more relevant documents (11) than system C+ (5), but due to the low fusion weight of T ($\beta_q = 0.25$, caused by a low AP value for T) 11 relevant documents were ranked so high (towards the end of the list) by fusion that they were cut off by the rank threshold (100) imposed by the ImageCLEF evaluation protocol. However, the achieved mean recall efficiency of 0.78 is certainly an important cause of improved retrieval performance of QAF with respect to each component system.

The utility of concept-based retrieval for QAF effectiveness is apparent from six queries (4, 11, 12, 16, 18, 25) where the fusion weight of component system T is displayed as 0.00, which actually means that $\beta_q < 0.005$. For these queries, text-based retrieval (T) had a poor average precision (for queries 4 and 11 AP was actually zero) and the achieved fusion retrieval performance is almost entirely due to concept-based retrieval. Note, however, that the C-utility values (both ω_C and ω_C^*) do not reach 1 for four of these queries (12, 16, 18, 25), because some of the relevant documents retrieved by component system C+ were also retrieved by T (although at much higher ranks). There is only one query (10), for which neither component system T nor C+ could retrieve any relevant document (the fusion weight was set to 0.5 in this case, because performance square weighting is undefined). Query-adaptive fusion almost always retained all relevant documents that were exclusively retrieved by the C+ system, which is recognized from the 32 queries where $\omega_C = \omega_C^*$.

Table 7.4: Per-query recall analysis of query-adaptive fusion Q(T,C+). In addition to recall-based measures (see main text), the number $|R|$ of judged relevant documents, fusion weights β_q, and average precision AP are presented.

Query	$\|R\|$	β_q	AP	ρ_F	ρ_F^*	φ	ω_C	ω_C^*
1	21	0.25	0.23	0.24	0.62	0.38	0.15	0.15
2	3	0.72	0.44	1.00	1.00	1.00	0.00	0.00
3	3	0.50	1.00	1.00	1.00	1.00	0.00	0.00
4	4	0.00	0.50	0.50	0.50	1.00	1.00	1.00
5	34	0.62	0.56	0.85	0.91	0.94	0.00	0.00
6	54	0.06	0.52	0.61	0.70	0.87	0.47	0.47
7	33	0.21	0.28	0.36	0.61	0.40	0.40	0.40
8	40	0.08	0.37	0.45	0.70	0.64	0.46	0.46
9	3	1.00	0.05	0.67	0.67	1.00	0.00	0.00
10	1	0.50	0.00	0.00	0.00	0.00	0.00	0.00
11	1	0.00	1.00	1.00	1.00	1.00	1.00	1.00
12	3	0.00	0.35	0.67	1.00	0.67	0.33	0.33
13	24	0.38	0.53	0.67	0.92	0.73	0.27	0.27
14	58	0.79	0.53	0.64	0.71	0.90	0.17	0.24
15	5	0.94	0.87	1.00	1.00	1.00	0.00	0.00
16	2	0.00	0.50	0.50	1.00	0.50	0.00	0.00
17	1	1.00	0.03	1.00	1.00	1.00	0.00	0.00
18	10	0.00	0.70	0.90	1.00	0.90	0.60	0.60
19	17	0.56	0.54	0.82	1.00	0.82	0.29	0.29
20	32	0.00	0.33	0.34	0.53	0.65	0.65	0.65
21	32	0.48	0.40	0.59	0.69	0.86	0.41	0.41
22	53	0.25	0.22	0.28	0.57	0.50	0.23	0.23
23	38	0.68	0.54	0.84	0.87	0.97	0.03	0.03
24	11	0.05	0.54	0.64	0.82	0.78	0.56	0.56
25	3	0.00	0.63	1.00	1.00	1.00	0.67	0.67
26	101	0.64	0.14	0.28	0.50	0.56	0.14	0.22
27	8	0.31	0.39	0.63	0.75	0.83	0.00	0.00
28	7	0.01	0.44	0.57	0.86	0.67	0.17	0.17
29	15	0.95	0.38	0.87	0.93	0.93	0.00	0.07
30	41	0.52	0.30	0.59	0.71	0.83	0.17	0.17
31	2	0.64	0.67	1.00	1.00	1.00	0.00	0.00
32	26	0.05	0.35	0.38	0.69	0.56	0.28	0.28
33	4	0.25	0.42	0.50	1.00	0.50	0.00	0.00
34	9	0.95	0.50	0.89	0.89	1.00	0.00	0.00
35	10	0.28	0.48	0.60	0.70	0.86	0.00	0.00
mean	20	–	0.45	0.65	0.80	0.78	0.24	0.25

The average C-utility values ($\omega_C = 0.24$ and $\omega_C^* = 0.25$), however, suggest that the overall contribution of concept-based retrieval to achieved QAF recall is minor. In fact, $\omega_C^* = 0$ for 13 queries, meaning that concept-based retrieval could not retrieve relevant documents that have not also been retrieved by the text-based component system. We conclude that the main contribution of concept-based retrieval to improved mean average precision of QAF is to help relevant documents rank lower (towards the top of the list).

7.4 Summary

This chapter proposed a multimodal retrieval framework applicable to MCR, that combines two well performing retrieval methods identified in previous chapters: text-based retrieval including query expansion and concept-based retrieval, which represents queries and documents by MeSH concepts. Ranking lists produced by the two component retrieval systems are combined by late fusion techniques that perform a linear combination of normalized scores of documents retrieved by component systems. For experiments, two late fusion methods were evaluated, one using fixed fusion weights for all queries (linear fusion), and one computing query-specific fusion weights from query performance predictions of component systems (query-adaptive fusion, QAF).

The focus of experimental evaluation was laid on estimating upper bounds of the retrieval performance achievable by the proposed framework on the given MCR dataset. Fusion experiments therefore combined the best performing text-based retrieval method identified in Chapter 5 (method T) and two best performing concept-based retrieval methods evaluated in Chapter 6: one uses a kNN classifier finding nearest neighbors by fulltext retrieval (method C), and the other uses ground-truth information to determine MeSH concepts from actually relevant documents (method C+). In light of the evaluation purpose, logistic score normalization and parameter optimization used the same query set as measuring retrieval performance.

Figure 7.5 depicts retrieval performance numbers achieved by evaluated fusion methods and compares them to the performance of component systems and fulltext retrieval. All tested fusion methods improved early and mean average precision over component systems, where the relative improvement depends on the performance of the concept-based component system: while fusion of practical component systems T and C displays only a small increase in MAP over T (3% for linear fusion, 9% for QAF), fusion of T and ideal component system C+ gives a more pronounced relative improvement over C+ (19% in MAP for linear fusion, 22% for QAF). The difference in effectiveness between linear and query-adaptive fusion is rather small, which may be caused by best-case optimizations explained in the previous paragraph.

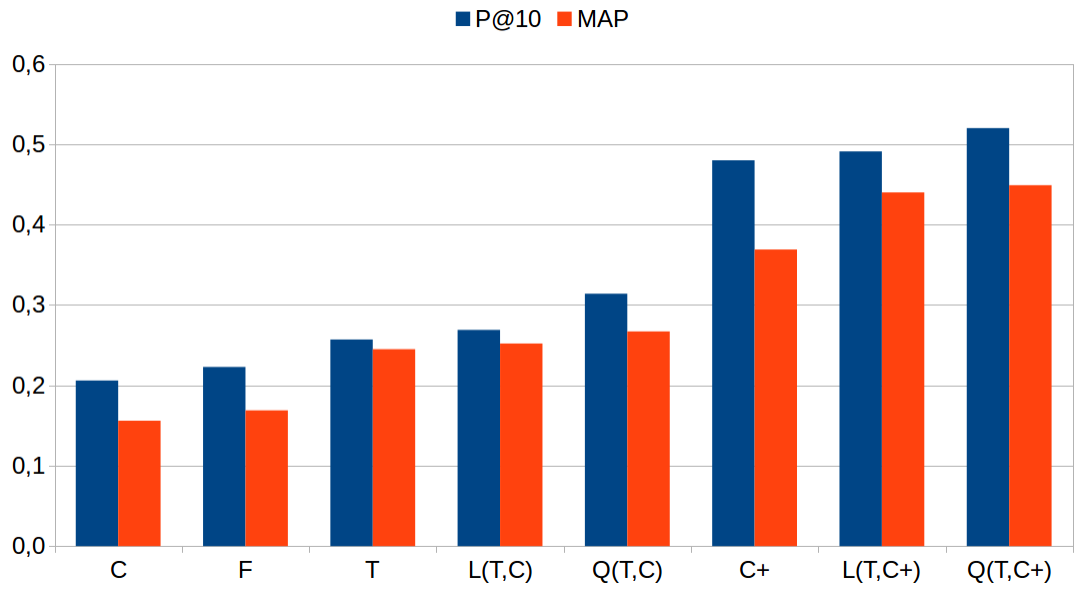


Figure 7.5: Retrieval performance of fulltext (F), improved text-based (T), concept-based (C), ideal concept-based (C+), and multimodal retrieval methods (linear fusion L, ideal query-adaptive fusion Q) on MCR dataset.

A per-query recall analysis performed for query-adaptive fusion of component systems T and C+ revealed that, for some queries, concept-based retrieval was able to find relevant documents that were not retrieved by the text-based system T, but the overall contribution of component system C+ to recall was rather low. Hence, the main contribution of concept-based retrieval to improved effectiveness of fusion was to move relevant documents closer to the top of the ranked list, thereby improving early and mean average precision.

Conducted experiments demonstrated an impressive potential for improvement of effectiveness when applying the proposed multimodal retrieval framework to the MCR dataset, compared to plain fulltext retrieval (0.169 MAP). Improvements range from 49% (0.252 MAP) for linear fusion of practical component systems T and C, to 166% (0.449 MAP) for ideal query-adaptive fusion of component systems T and C+. To utilize this potential with practical solutions, further research needs to address the problems of constructing more effective concept-based retrieval systems and query-adaptive fusion methods.

CHAPTER 8

Concluding Remarks

The purpose of this chapter is threefold: (1) it provides an overview of experimental results covered by this thesis (Section 8.1), followed by a discussion of limitations of evaluation caused by characteristics of the MCR dataset (Section 8.2); (2) proposed methods and achieved results are related to the research objectives and contributions stated in Chapter 1, leading to the conclusion of this thesis (Section 8.3); and (3) avenues for further research emerging from this thesis are identified (Section 8.4).

8.1 Summary of Results

This thesis proposed and evaluated automatic methods addressing the problem of medical case retrieval (MCR) with the aim of utilizing multiple modalities (text, images, and terms of a controlled biomedical vocabulary) representing medical case descriptions and case queries such that MCR effectiveness improves over plain fulltext retrieval. Most experiments were conducted on the MCR dataset described in Chapter 3, where a relevant preprocessing task for images found in scientific articles and medical case descriptions has been addressed: the automatic detection and separation of compound figures. To support the hypothesis that biomedical concepts provided by controlled vocabularies can help to improve MCR effectiveness, methods for automatically mapping textual and visual modalities to biomedical concepts were evaluated in Chapter 4. Retrieval methods investigated in subsequent chapters utilized biomedical concepts to improve text-based retrieval (Chapter 5), perform concept-based retrieval (Chapter 6), and combine these retrieval methods by late fusion (Chapter 7).

A novel method for compound figure separation (CFS) has been proposed and evaluated on two public datasets. The proposed automatic method was shown to be more effective than existing automatic and semi-automatic techniques that used the same datasets for evaluation. Additionally, a novel compound figure classifier (CFC) has been proposed that turned out to be not as effective as other known complex algorithms, but its efficiency enables bulk processing of large datasets that was demonstrated on 300k images of the MCR dataset. Furthermore, the sequential application of CFC followed

by CFS showed that CFC accuracy is not critical for the effectiveness of the CFC-CFS chain. CFC-CFS processing of the MCR dataset resulted in 800k images and allowed to estimate the dataset's compound figure rate experimentally as 50%.

We described approaches for three types of concept mapping techniques that use textual, visual or both modalities of case descriptions to assign biomedical concepts of the Medical Subject Headings (MeSH) thesaurus. The effectiveness of text-to-concept mapping was measured on the MCR dataset in two different aspects: the ability of algorithms to reproduce manually annotated MeSH terms (evaluated in Chapter 4), and their effectiveness for concept-based retrieval (in Chapter 6). Due to missing ground-truth information for images of the MCR dataset, image-to-concept mapping could be evaluated by concept-based retrieval only. The evaluation of concept mapping by multi-view learning has been postponed to future work due to its high implementation cost.

From four tested text-to-concept mapping systems, a nearest-neighbor (kNN) classifier obtaining MeSH concepts from pseudo-relevant documents retrieved by fulltext outperformed other algorithms by large margins in both evaluations. A class of novel proposed text-to-concept mapping algorithms based on string matching, in particular the BinDist algorithm, displayed similar or better effectiveness than the well-known MetaMap system used by the U.S. National Library of Medicine to aid manual MeSH annotation of scientific biomedical articles. However, whereas all other tested algorithms, including the kNN classifier, are limited to short input documents, string matching algorithms can be applied to large datasets of long documents due to a substantially better run-time efficiency. In fact, the BinDist algorithm was applied to produce automatic MeSH annotations of all articles in the MCR dataset, which served to enrich the concept-to-document index for all concept-based retrieval experiments.

Lead by the success of kNN classifiers for text-to-concept mapping, we evaluated three variants of visual kNN classifiers, which differ in the way how pseudo-relevant images (nearest neighbors) are retrieved from the image index. Concept-based retrieval performance of all variants was inferior to the corresponding performance of the BinDist algorithm, and the worst performance was displayed by the simplest visual kNN variant, which employed content-based image retrieval to determine nearest neighbors. The variant applying concept-based retrieval to the image index, where MeSH concepts of images were extracted from image captions by the BinDist algorithm, displayed the best performance among visual kNN classifiers, but it does not utilize content-based image features at all.

Given the fact that none of the evaluated concept-based retrieval methods could improve over plain fulltext retrieval (16.9% MAP) on the MCR dataset, extensive experiments trying to improve text-based retrieval by query expansion and document expansion were conducted (Chapter 5). Query expansion by local feedback selecting

unigrams and bigrams (2-word-sequences) from pseudo-relevant documents turned out to cause the highest performance gains, and preceeding local feedback with query expansion by MeSH terms extracted from the query text using our string matching algorithms delivered the best retrieval performance in our experiments (24.5% MAP). Document expansion by MeSH terms, on the other hand, could not consistently improve other text-based retrieval methods.

Evaluation of the proposed multimodal retrieval framework (Chapter 7) focused on exploring the potential of further improvements on the given MCR dataset by combining text-based and concept-based retrieval through late fusion techniques. To this end, some "ideal" retrieval and fusion methods were considered that allow to estimate upper bounds of the effectiveness of practical systems by experimental evaluation. An ideal concept-based retrieval method (C+) obtaining MeSH concepts from actually relevant documents retrieved by fulltext (according to ground-truth judgments) indeed achieved a remarkable retrieval performance of 36.9% MAP, demonstrating that concept-based retrieval has the potential to improve over text-based retrieval if better concept-mapping algorithms can be designed. By combining ideal concept-based retrieval with the best practical text-based retrieval method (T) identified earlier, mean average precision could be further increased to 44.9% MAP (166% increase over fulltext retrieval). When comparing linear score fusion with an ideal query-adaptive fusion method that determines per-query fusion weights from the actual query performance of component systems, we saw that the difference in effectiveness was rather modest: 44.0% vs. 44.9% MAP for fusion of T and C+, and 25.2% vs. 26.7% MAP for fusion of T with the best practical concept-based system (textual kNN classifier, named C). The best tested practical fusion system, implemented by linear fusion of T and C component systems, achieved with 25.2% MAP an increase of 49% in effectiveness over fulltext retrieval. A chart comparing the effectiveness of mentioned retrieval methods has been presented in Fig. 7.5 (on page 142).

8.2 Limitations of MCR Dataset

Expressiveness, generalizability, and extent of evaluations conducted for this thesis were limited by available relevance judgments and manual MeSH annotations for the MCR dataset (see Section 3.1) used for most experiments. Limitations caused by relevance judgments arise from the pooling method used to select documents that were presented to medical experts for judging. Manual MeSH annotations tend to be incomplete [212, p. 153] and biased by the domain of expertise of human annotators.

Following the well-known TREC-style evaluation methodology [219], the problem of selecting documents for manual relevance judgments has been addressed by *pooling*

Table 8.1: Number of judged documents per query retrieved from the MCR dataset by methods used in Chapter 7. The total number of retrieved documents per query was kept fixed at 100.

Method	*Retrieved judged documents*		
	min	max	average
T	36	83	64.7
L(T,C)	36	83	64.7
Q(T,C)	15	83	56.3
L(T,C+)	31	76	55.6
Q(T,C+)	5	76	36.9
C	15	78	34.1
C+	2	56	18.0

documents retrieved by several different retrieval systems from the MCR dataset, separately for each query. As presented in Table 3.1 (on page 36), this procedure resulted in 429 judged documents per query on average for the ImageCLEF MCR dataset, with about 20 judged relevant documents per query.

Whereas such a pooling strategy provides satisfactory quality of evaluation for the retrieval systems used for pooling, it may fail to produce meaningful results for novel retrieval systems whose set of retrieved documents for a given query has only a small intersection with the set of judged documents. In fact, evaluation measures based on precision and recall (including mean average precision) treat retrieved non-judged documents as *not relevant* for a given query, although some of them may actually turn out to be relevant if judged by a human expert. Hence, the intersection size between the sets of retrieved and judged documents for a query may serve as an indicator for quality of evaluation based on precision and recall. The smaller the intersection size, the stronger is the need for additional human relevance judgments to adequately evaluate a novel retrieval system.

Table 8.1 presents some statistics on the number of retrieved judged documents per query for retrieval methods used in experiments of Chapter 7. Most retrieval systems involving text-based methods display an average judged document rate of more than 55%, which can be explained by the fact that documents selected for judgment (by the pooling procedure) were obtained primarily from text-based retrieval systems. Concept-based retrieval systems (C and C+), on the other hand, retrieved much less judged documents, causing an accordingly lower quality of evaluation. Another consequence of this low judged document rate is that the probability of finding relevant documents among retrieved non-judged ones is much higher for concept-based retrieval methods than for text-based ones.

A related problem occurs for queries whose relevance judgments label only very few documents as relevant. Table 7.4 (on page 140) shows that the MCR dataset contains three queries with only one judged relevant document. For such queries the maximum achievable precision at 10 (P@10) value is 0.1, degrading the average P@10 value over all queries accordingly. Although this effect is not relevant for average precison (AP may be 1 even for queries with one judged relevant document), evaluation quality is poor, because it fully depends on the ability of the system under test to retrieve a single judged relevant document.

While properties of available relevance judgments limit the quality of retrieval evaluation, manual MeSH annotations determine the quality of concept mapping evaluations, as conducted in Chapter 4. Experiments measured the ability of text-to-concept mapping algorithms to reproduce manual MeSH annotations of documents in the MCR dataset. If manual (ground-truth) annotations are incomplete or biased by the domain of expertise of human annotators, automatic concept mapping algorithms may produce MeSH concepts that are not (closely) related to available manual annotations, but actually relevant for the document. Text classification evaluation measures can therefore not capture the "true" classification accuracy of tested algorithms. Note that hierarchical evaluation measures (see Section 4.5.1) may alleviate this problem to some extent if concepts produced by automatic concept mapping are closely related to ground-truth annotations, but generally they will not be able to compensate incompleteness or bias of manual MeSH annotations. Moreover, missing ground-truth MeSH annotations of images contained in documents of the MCR dataset prohibited the evaluation of classification performance of image-to-concept mapping algorithms.

Consequently, to overcome these limitations for evaluation of concept mapping and retrieval systems, additional ground-truth annotations (both relevance judgments and MeSH annotations) are needed, either for the existing MCR dataset or for an additional dataset acquired or built in future work.

8.3 Conclusion

The main research objective of this thesis was to develop and evaluate multimodal methods for medical case retrieval (MCR) that improve over plain fulltext retrieval, thereby testing the hypothesis that biomedical concepts taken from a controlled vocabulary can be the main cause of improvement. Although practical concept-based retrieval methods were confirmed to be inferior to plain fulltext retrieval (as already known from literature [212]), the utilization of biomedical concepts for query expansion in text-based retrieval systems was shown to be effective and, together with pseudo-relevance feedback for query expansion, increased mean average precision (MAP) by 45% compared to fulltext retrieval. The combination of improved text-based retrieval with practical

concept-based retrieval by late fusion techniques could even add to this performance gain by another 4–13%, depending on the fusion method. The null hypothesis, that biomedical concepts *cannot* help to improve MCR over fulltext retrieval, can therefore be rejected with high confidence and justifies further research in this direction.

To prepare the way for future work, the potential of concept-based retrieval and advanced late fusion techniques has been evaluated by considering "ideal" methods that use ground-truth information from relevance judgments to simulate parts of the proposed multimodal retrieval framework with maximal effectiveness. Experiments showed an increase by more than 160% in MAP for the fusion of ideal concept-based retrieval with improved text-based retrieval, suggesting that there is room for another substantial improvement of concept-based techniques in the future.

The detailed research objectives derived from the main research goal in Section 1.3 led to a number of contributions of this thesis (see also Section 1.5). For preprocessing images contained in case descriptions and targeting research objective O1, novel automatic methods for compound figure classification and separation have been proposed and evaluated [209], that improve over state-of-the-art techniques while allowing an efficient processing rate of 12 images per second in a prototype software implementation. Text-to-concept and image-to-concept mapping algorithms have been proposed and compared with respect to their effectiveness for concept-based retrieval on the MCR dataset (objectives O2 and O4). Extensive experimental evaluation of different text-based retrieval methods has been conducted, resulting in the identification of effective method combinations that achieve state-of-the-art retrieval performance on the MCR dataset without relying on external text corpora (objective O3). Finally, a novel multimodal retrieval framework combining text-based and concept-based retrieval methods has been proposed, that was demonstrated to outperform state-of-the-art retrieval methods on the MCR dataset (objective O5).

8.4 Further Work

During work on this thesis, several avenues and possibilities to extend our work have been identified. The following sections provide an overview of ideas for further research, grouped by topics that roughly correspond to previous chapters of this thesis. Accordingly, further work could include research on modality classification of images contained in medical case descriptions (Section 8.4.1), on extended evaluation of concept mapping algorithms (Section 8.4.2) and text-based retrieval (Section 8.4.3), on practical query-adaptive fusion methods (Section 8.4.4), on retrieval in multi-view latent space (Section 8.4.5), and on improving MCR by learning from users (Section 8.4.6). Most suggestions for further work will require an effort of a few person weeks or months,

but some may well extend to one person year (e.g. building a new dataset). If applicable, further work topics are ordered with respect to increasing expected cost of implementation within the following sections.

8.4.1 Image Preprocessing

Case queries of the ImageCLEF MCR dataset contain only diagnostic images relevant for a patient case. Documents of the dataset, however, contain arbitrary images found in scientific biomedical articles, including diagrams, charts, and other non-medical images. Although the ratio of diagnostic images contained in the ImageCLEF MCR dataset may be smaller than in a typical collection of case descriptions found in health-care institutions, the automatic classification of diagnostic images may help to retrieve medical images that are relevant for a given case query.

The task of *modality classification* of medical images has been posed in multiple challenges by the ImageCLEF evaluation campaign between 2010 and 2015 [88, 90]. A class hierarchy containing 38 classes of diagnostic images and general biomedical illustrations has been defined, and annotated datasets were provided to enable evaluation of classifiers submitted by participants. In 2015, the task was cast as a multi-label classification problem of compound images, and best results were achieved by an approach based on deep convolutional neural networks (see [90]). However, none of the participants applied state-of-the-art multi-label classification techniques [246]. In 2013, best single-label classification results were achieved by classical classifiers (SVM, kNN) using hand-crafted image feature extractors [88], deep learning solutions had not yet been applied.

A promising and necessary direction for future work in preprocessing of medical case images is therefore the application of *deep learning* techniques [19, 109, 117] to both modality classification and compound figure detection, which may enable an effective filtering of images prior to indexing for retrieval. Furthermore, an automatic classification of body parts represented in medical images, as defined by the IRMA code [120], may provide additional concepts that could be used for concept-based retrieval.

8.4.2 Concept Mapping

Evaluation of concept mapping algorithms conducted for this thesis had some restrictions due to different reasons (scope of work, time constraints, and limitations of the MCR dataset). Further work could therefore extend experimental evaluation in the following aspects, ordered roughly by increasing cost of effort:

- Evaluate the effectiveness of *IdfBinDist* and *IdfCovDist* string matching algorithms (see Section 4.2.3) by the same type of experiments as conducted in Chapters 4 and 6.

- Evaluate image-to-concept mapping algorithms (Section 6.2.2) with optimized parameters, additional image features (e.g. a compact descriptor for radiology images [39]), and multimodal indexing techniques (e.g. global feature mapping [188]).

- Consider the utilization of other biomedical vocabularies and ontologies (e.g. UMLS, SNOMED-CT) for concept mapping, as mentioned in the introduction to Chapter 4.

- Evaluate concept mapping by multi-view learning described in Section 4.4, using the available implementation of the approach by Xu et al. [235]. Preliminary trials indicated problems with convergence of the optimization algorithm and with effectiveness due to the large dimensionality of concept space, as explained in Section 4.4.2.

- Perform a study of manual MeSH annotations testing the hypothesis that MeSH annotations selected by human domain experts belong to a certain (domain-dependent) subset of all available MeSH concepts, and that certain levels within MeSH subtrees are preferred. The hypothesis is motivated by the work of Trieschnigg [212, p. 153], who found that between 34% and 58% of MeSH terms that were predicted by automatic concept mapping, but did not correspond to manual annotations, were actually relevant to documents.

- Build or acquire an MCR dataset with more complete ground-truth MeSH annotations of both documents and images, enabling a more powerful evaluation of automatic concept mapping algorithms. To enhance the expressiveness of concept-based retrieval results, additional relevance judgments selected by appropriate pooling (and performed by medical experts) are needed.

- Apply deep learning techniques [19, 109, 117] to the problem of concept mapping, where textual, visual, or both modalities of case descriptions may be used as input to a deep neural network. Recent advances in image caption generation [217] may serve as a starting point for developing a concept mapping approach based on deep learning. Note, however, that the acquisition or development of an appropriate (large) training dataset may be needed.

8.4.3 Text-Based Retrieval

As noted in Section 5.4, additional efforts could be spent on an improved evaluation of text-based retrieval methods, as well as on applying different advanced retrieval methods known to work for general information retrieval. Such efforts include, again ordered by increasing expected cost of implementation:

- Utilizing document structure (e.g. title, abstract, and image captions) to weight document terms differently for relevance ranking.

- Cross-validating parameter optimization using a different (e.g. genetic) algorithm, thereby resolving the dependency on the heuristic application of the SPSA algorithm used in experiments of Chapter 5.

- Building or acquiring a second MCR dataset (could be the same as mentioned in Section 8.4.2) allowing to assess the generalization ability of tested retrieval methods more confidently.

- Applying more sophisticated query expansion methods (see Section 2.2.3), including the utilization of external corpora or text categorization based on machine learning [182].

8.4.4 Query-Adaptive Fusion

Experiments in Chapter 7 considered ideal query-adaptive fusion (QAF) only to obtain upper bounds for the effectiveness of practical QAF methods. Further work could therefore measure the retrieval performance delivered by practical QAF systems and compare it to obtained upper bounds. The investigation of practical QAF methods could proceed in two phases:

1. Keep the performance square weighting scheme [230] used in experiments of Chapter 7 and apply known methods [56, 55, 187, 252] to predict the query performance of component systems. As a starting point, the method by Cummins et al. [56] estimates query performance from the standard deviation of scores assigned by component systems for a query-specific number of top-ranked documents, and is easy to implement.

2. Consider other performance weighting schemes (see [229]) or different query-adaptive fusion strategies [106] that may be beneficial for fusion of text-based and concept-based retrieval.

The reliability of evaluation results could be improved (with respect to results obtained in Chapter 7) by either using a separate dataset for score normalization training and parameter optimization, or by applying cross-validation to the query set (given the 35 queries of the MCR dataset, a 5-fold cross-validation would be suitable).

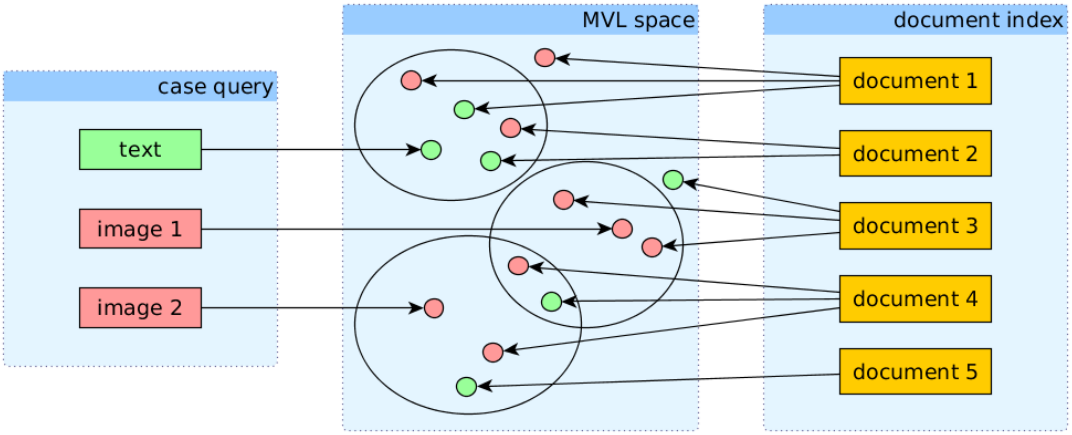


Figure 8.1: Direct document retrieval for a given case query in multi-view latent (MVL) space.

8.4.5 Retrieval in Multi-View Latent Space

The multi-view subspace learning approach applied to concept mapping (Section 4.4) may also be used for direct retrieval in the learned multi-view latent (MVL) space. Such an approach would represent an alternative to the multimodal retrieval framework proposed in Section 7.1.

Multi-view subspace learning algorithms learn mappings from each view (textual or visual representations of case descriptions) to a common (low-dimensional) MVL space such that different views of the same source instance (case description) are mapped to nearby points in MVL space. Based on the assumption that nearby points in MVL space represent semantically similar case descriptions, retrieval could be implemented by finding nearest neighbors of a point in MVL space that represents a given case query. The same technology used to implement content-based image retrieval (e.g. by LIRE[1]) can be applied to retrieve nearest neighbors in MVL space efficiently.

Figure 8.1 illustrates the anticipated retrieval method in MVL space. At indexing time, textual (green) and visual (red) views of case descriptions (documents) are mapped to points in MVL space, which are stored together with document references in an index supporting efficient retrieval of nearest neighbors in MVL space. Given a case query, the same mapping is applied to obtain query points in MVL space, and their nearest neighbors (indicated by ellipses) determined using the index represent candidate documents for producing the ranked result list.

How the ranking of candidate documents is exactly achieved, is the subject of future work. Some suggestions are: (1) aggregate distances of candidate points in MVL space

[1] `http://www.lire-project.net/`

representing the same document to their nearest query point; (2) utilize biomedical concepts associated with points in MVL space (as available with the selected multi-view learning approach [235]); (3) apply a learning-to-rank approach [125].

8.4.6 Learning from Users

Since users of a medical case retrieval system are likely to be medical experts, learning from them may provide an alternative approach to improving retrieval effectiveness that goes well beyond the scope of this thesis and may be the subject of future work. A classical way of utilizing feedback of users for the retrieval process is *relevance feedback* (RF).

Relevance feedback has been an active research field in multimedia information retrieval for several decades [250, 57], because it attempts to address the semantic gap problem by incorporating relevance judgments from users. Algorithmic approaches to RF can be categorized as *short-term learning* and *long-term learning* techniques [250], depending on the desired effect of user feedback on retrieval results: short-term learning affects the current query only [112], whereas long-term learning aims at improving retrieval performance for future queries [52]. More recent approaches include a probabilistic RF framework processing multiple image queries consisting of both positive and negative samples [8] for short-term learning, and a semi-supervised long-term learning algorithm [237].

Many RF methods utilize relevance judgments of users as additional training data for machine learning. Depending on whether also unlabeled training data are used for learning, *inductive* (using only labeled training data) and *transductive* methods can be distinguished. A prominent technique for transductive RF is manifold-ranking [86], a more recent extension using random walks has been proposed by Rota Bulò et al. [170].

RF learning methods have to cope with the small sample size problem, because the number of training samples provided by relevance feedback is usually too small to reliably improve prediction performance for most learning algorithms. It is therefore desireable that the system selects samples for relevance feedback that, when labeled by the user, yield maximal performance improvement for the learning algorithm with respect to some optimization criterion. This is exactly the problem addressed by the *active learning* literature [221, 183]. However, choosing the most informative samples will most likely not coincide with the most positive samples the user is interested in, so active learning techniques applied to iterative short-term learning often rely on the user's patience [250]. Active learning may therefore be more interesting for long-term learning.

APPENDIX

A Implementation Details

A.1 Parameters of Compound Figure Separation

The proposed CFS algorithm (Section 3.2.2) takes 17 internal parameters listed in Table A.1. Parameters marked by * use units of image width, height, or area, depending on the parameter and processing direction (horizontal or vertical).

Initial parameter values were chosen manually by looking at the results produced for a few training images. They were used during participation in ImageCLEF 2015 [208]. For parameter optimization, the CFS algorithm was evaluated for various parameter combinations on the ImageCLEF 2015 CFS training dataset (3,403 compound images, 14,531 ground-truth subfigures) using the evaluation tool provided by ImageCLEF organizers. Due to the number of parameters and the run time of a single evaluation run (about 17 minutes), a grid-like optimization evaluating all possible parameter combinations in a certain range was not feasible. Instead, we applied a hill-climbing optimization strategy to locate the region of a local maximum and then used grid optimization in the neighborhood of this maximum.

More precisely, we defined up to five different values per parameter, including the initial values, on a linear or logarithmic scale, depending on the parameter. Then a set of parameter combinations was generated where only one parameter was varied at a time and all other parameters were kept at their initial values, resulting in a feasible number of parameter combinations to evaluate (linear in the number of parameters). After measuring accuracy on the training set, the most effective value of each parameter was chosen as its new *optimal* value. For parameters whose optimal values differed from the initial ones, the range was centered around the optimal value. Other parameters were fixed at their latest value. The procedure was repeated until accuracy improved by no more than 5%, which happened after three iterations. Finally, after sorting parameter combinations by achieved accuracy, the five most effective parameters were chosen for grid optimization, where only two “nearly optimal" values (including the latest optimal value) per parameter were selected.

Table A.1: Internal parameters of proposed CFS algorithm.

Parameter	*Initial*	*Optimal*	*Meaning*
Main algorithm			
`classifier_model`	first	greedy	first, majority, unanimous, or greedy (see Section 3.2.2.1)
`decision_threshold`	0.5	0.1	minimal illustration class probability to decide in favor of band-based separator detection
`mindim`	50	200	minimal image dimension (pixels) to apply separator detection to
`elim_area`	0	0.03	area threshold to eliminate small bounding boxes*
Edge-based separator detection			
`edge_maxdepth`	10	10	maximal recursion depth
`edge_sobelthresh`	0.05	0.02	threshold for Sobel edge detector
`edge_houghratio_min`	0.25	0.2	minimal ratio of Hough values for peak selection
`edge_houghratio_base`	1.2	1.5	base of recursion depth dependency for Hough peak selection
`edge_maxdistvar`	0.0001	0.1	maximal variance of separator distances for regularity criterion*
`edge_gapratio`	0.2	0.3	gap threshold for edge filling*
`edge_lenratio`	0.05	0.03	length threshold for edge filling*
`edge_minseplength`	0.7	0.5	minimal separator length*
`edge_minborderdist`	0.1	0.05	minimal distance of separators from border*
Band-based separator detection			
`band_maxdepth`	2	4	maximal recursion depth
`band_minsepwidth`	0.03	0.0001	minimal width of separator bands*
`band_maxdistvar`	0.0003	0.2	maximal variance of separator distances for regularity criterion*
`band_minborderdist`	0.1	0.01	minimal distance of separators from border*

The effect of parameter optimization was surprisingly strong: whereas the initial parameter configuration achieved an accuracy of 43.5% on the training set, performance increased to 84.5% after hill-climbing optimization, and finished at 85.5% after grid optimization.

List of Figures

List of Tables

Bibliography

[1] Aamodt, A., Plaza, E.: Case-based reasoning: Foundational issues, methodological variations, and system approaches. AI Communications **7**(1), 39–59 (1994)

[2] Abdou, S., Savoy, J.: Searching in Medline: Query expansion and manual indexing evaluation. Inf. Process. Manage. **44**(2), 781–789 (2008). DOI 10.1016/j.ipm.2007.03.013

[3] Agirre, E., Di Nunzio, G.M., Mandl, T., Otegi, A.: CLEF 2009 ad hoc track overview: Robust-WSD task. In: Proceedings of the 10th CLEF Conference on Multilingual Information Access Evaluation: Text Retrieval Experiments, CLEF'09, pp. 36–49. Springer-Verlag, Berlin, Heidelberg (2009)

[4] Amati, G., Carpineto, C., Romano, G.: Comparing weighting models for monolingual information retrieval. In: Comparative Evaluation of Multilingual Information Access Systems, Proc. 4th Workshop of the Cross-Language Evaluation Forum, CLEF 2003, pp. 310–318. Springer (2004)

[5] Amati, G., Van Rijsbergen, C.J.: Probabilistic models of information retrieval based on measuring the divergence from randomness. ACM Trans. Inf. Syst. **20**(4), 357–389 (2002). DOI 10.1145/582415.582416

[6] ao, A.M., Martins, F., ao Magalhães, J.: Multimodal medical information retrieval with unsupervised rank fusion. Computerized Medical Imaging and Graphics **39**, 35–45 (2015). DOI 10.1016/j.compmedimag.2014.05.006. URL `http://www.sciencedirect.com/science/article/pii/S0895611114000664`

[7] Apostolova, E., You, D., Xue, Z., Antani, S., Demner-Fushman, D., Thoma, G.R.: Image retrieval from scientific publications: Text and image content processing to separate multipanel figures. Journal of the American Society for Information Science and Technology **64**(5), 893–908 (2013). DOI 10.1002/asi.22810

[8] Arevalillo-Herráez, M., Ferri, F.J., Domingo, J.: A naive relevance feedback model for content-based image retrieval using multiple similarity measures. Pattern Recogn. **43**(3), 619–629 (2010). DOI 10.1016/j.patcog.2009.08.010

[9] Arguello, J., Elsas, J.L., Callan, J., Carbonell, J.G.: Document representation and query expansion models for blog recommendation. In: E. Adar, M. Hurst, T. Finin, N. Glance, N. Nicolov, B. Tseng (eds.) Proc. 2nd Int. Conf. Weblogs and Social Media, pp. 10–18. AAAI Press (2008)

[10] Aronson, A., Bodenreider, O., Chang, H., Humphrey, S., Mork, J., Nelson, S., Rindflesch, T., Wilbur, W.: The NLM indexing initiative. Proc. AMIA Symposium pp. 17–21 (2000). URL http://europepmc.org/articles/PMC2243970

[11] Aronson, A.R., Lang, F.M.: An overview of MetaMap: historical perspective and recent advances. Journal of the American Medical Informatics Association **17**(3), 229–236 (2010). DOI 10.1136/jamia.2009.002733

[12] Baeza-Yates, R., Ribeiro-Neto, B.: Modern Information Retrieval, 2nd edn., chap. 1 (Introduction), pp. 1–19. Addison-Wesley Publishing Company, USA (2011)

[13] Baeza-Yates, R., Ribeiro-Neto, B.: Modern Information Retrieval, 2nd edn. Addison-Wesley Publishing Company, USA (2011)

[14] Baeza-Yates, R., Ribeiro-Neto, B.: Modern Information Retrieval, 2nd edn., chap. 8 (Text Classification), pp. 281–335. Addison-Wesley Publishing Company, USA (2011)

[15] Bast, H., Majumdar, D., Weber, I.: Efficient interactive query expansion with complete search. In: Proceedings of the Sixteenth ACM Conference on Conference on Information and Knowledge Management, CIKM '07, pp. 857–860. ACM, New York, NY, USA (2007). DOI 10.1145/1321440.1321560

[16] Batet, M., Sánchez, D., Valls, A.: An ontology-based measure to compute semantic similarity in biomedicine. Journal of Biomedical Informatics **44**(1), 118–125 (2011)

[17] Begum, S., Ahmed, M., Funk, P., Xiong, N., Folke, M.: Case-based reasoning systems in the health sciences: A survey of recent trends and developments. Systems, Man, and Cybernetics, Part C: Applications and Reviews, IEEE Transactions on **41**(4), 421–434 (2011). DOI 10.1109/TSMCC.2010.2071862

[18] Bengio, Y.: Learning deep architectures for AI. Found. Trends Mach. Learn. **2**(1), 1–127 (2009). DOI 10.1561/2200000006

[19] Bengio, Y., Courville, A., Vincent, P.: Representation learning: A review and new perspectives. Pattern Analysis and Machine Intelligence, IEEE Transactions on **35**(8), 1798–1828 (2013). DOI 10.1109/TPAMI.2013.50

[20] Bergstra, J., Bengio, Y.: Random search for hyper-parameter optimization. J. Mach. Learn. Res. **13**, 281–305 (2012)

[21] Beygelzimer, A., Kakade, S., Langford, J.: Cover trees for nearest neighbor. In: Proceedings of the 23rd International Conference on Machine Learning, ICML '06, pp. 97–104. ACM, New York, NY, USA (2006). DOI 10.1145/1143844.1143857

[22] Bhogal, J., Macfarlane, A., Smith, P.: A review of ontology based query expansion. Inf. Process. Manag. **43**(4), 866–886 (2007). DOI 10.1016/j.ipm.2006.09.003

[23] Bickel, S., Scheffer, T.: Multi-view clustering. In: Proceedings of the Fourth IEEE International Conference on Data Mining, ICDM '04, pp. 19–26. IEEE Computer Society, Washington, DC, USA (2004)

[24] Bishop, C.M.: Pattern Recognition and Machine Learning, chap. 1.5 (Decision Theory), pp. 38–47. Springer, Secaucus, NJ, USA (2006)

[25] Bishop, C.M.: Pattern Recognition and Machine Learning (Information Science and Statistics). Springer-Verlag New York, Inc., Secaucus, NJ, USA (2006)

[26] Blum, A., Mitchell, T.: Combining labeled and unlabeled data with co-training. In: Proceedings of the Eleventh Annual Conference on Computational Learning Theory, COLT' 98, pp. 92–100. ACM, New York, NY, USA (1998). DOI 10.1145/279943.279962

[27] Bosch, A., Zisserman, A., Munoz, X.: Representing shape with a spatial pyramid kernel. In: Proceedings of the 6th ACM International Conference on Image and Video Retrieval, CIVR '07, pp. 401–408. ACM, New York, NY, USA (2007). DOI 10.1145/1282280.1282340

[28] Bougeard, S., Hanafi, M., Qannari, E.: Continuum redundancy–PLS regression: A simple continuum approach. Computational Statistics and Data Analysis **52**(7), 3686–3696 (2008). DOI 10.1016/j.csda.2007.12.007

[29] Brefeld, U., Gärtner, T., Scheffer, T., Wrobel, S.: Efficient co-regularised least squares regression. In: Proceedings of the 23rd International Conference on Machine Learning, ICML '06, pp. 137–144. ACM, New York, NY, USA (2006). DOI 10.1145/1143844.1143862

[30] Brefeld, U., Scheffer, T.: Co-EM support vector learning. In: Proceedings of the Twenty-first International Conference on Machine Learning, ICML '04, pp. 16–23. ACM, New York, NY, USA (2004). DOI 10.1145/1015330.1015350

[31] Cao, G., Nie, J.Y., Gao, J., Robertson, S.: Selecting good expansion terms for pseudo-relevance feedback. In: Proceedings of the 31st Annual International ACM SIGIR Conference on Research and Development in Information Retrieval, SIGIR '08, pp. 243–250. ACM, New York, NY, USA (2008). DOI 10.1145/1390334.1390377

[32] Carpineto, C., de Mori, R., Romano, G., Bigi, B.: An information-theoretic approach to automatic query expansion. ACM Trans. Inf. Syst. **19**(1), 1–27 (2001). DOI 10.1145/366836.366860

[33] Carpineto, C., Osiński, S., Romano, G., Weiss, D.: A survey of web clustering engines. ACM Comput. Surv. **41**(3), 17:1–17:38 (2009). DOI 10.1145/1541880.1541884

[34] Carpineto, C., Romano, G.: A survey of automatic query expansion in information retrieval. ACM Comput. Surv. **44**(1), 1:1–1:50 (2012). DOI 10.1145/2071389.2071390

[35] Carpineto, C., Romano, G., Giannini, V.: Improving retrieval feedback with multiple term-ranking function combination. ACM Trans. Inf. Syst. **20**(3), 259–290 (2002). DOI 10.1145/568727.568728

[36] Caruana, R., Niculescu-Mizil, A.: An empirical comparison of supervised learning algorithms. In: Proceedings of the 23rd International Conference on Machine Learning, ICML '06, pp. 161–168. ACM, New York, NY, USA (2006). DOI 10.1145/1143844.1143865

[37] Chang, A.A., Heskett, K.M., Davidson, T.M.: Searching the literature using Medical Subject Headings versus text word with PubMed. The Laryngoscope **116**(2), 336–340 (2006). DOI 10.1097/01.mlg.0000195371.72887.a2

[38] Chang, Y., Ounis, I., Kim, M.: Query reformulation using automatically generated query concepts from a document space. Inf. Process. Manage. **42**(2), 453–468 (2006). DOI 10.1016/j.ipm.2005.03.025

[39] Chatzichristofis, S., Boutalis, Y.: Content based radiology image retrieval using a fuzzy rule based scalable composite descriptor. Multimedia Tools and Applications **46**, 493–519 (2010). DOI 10.1007/s11042-009-0349-x

[40] Chatzichristofis, S.A., Boutalis, Y.S.: CEDD: Color and edge directivity descriptor: A compact descriptor for image indexing and retrieval. In: A. Gasteratos, M. Vincze, J.K. Tsotsos (eds.) Computer Vision Systems, *Lecture Notes in Computer Science*, vol. 5008, pp. 312–322. Springer (2008). DOI 10.1007/978-3-540-79547-6_30

[41] Chatzichristofis, S.A., Boutalis, Y.S.: FCTH: fuzzy color and texture histogram - a low level feature for accurate image retrieval. In: Proceedings of the 2008 Ninth International Workshop on Image Analysis for Multimedia Interactive Services, WIAMIS '08, pp. 191–196. IEEE Computer Society, Washington, DC, USA (2008). DOI 10.1109/WIAMIS.2008.24

[42] Chen, N., Zhu, J., Xing, E.P.: Predictive subspace learning for multi-view data: a large margin approach. In: J.D. Lafferty, C.K.I. Williams, J. Shawe-Taylor, R.S.

Zemel, A. Culotta (eds.) Advances in Neural Information Processing Systems 23, pp. 361–369. Curran Associates, Inc. (2010)

[43] Chen, Y., Wang, L., Wang, W., Zhang, Z.: Continuum regression for cross-modal multimedia retrieval. In: 19th IEEE International Conference on Image Processing, pp. 1949–1952 (2012). DOI 10.1109/ICIP.2012.6467268

[44] Chhatkuli, A., Foncubierta-Rodríguez, A., Markonis, D., Meriaudeau, F., Müller, H.: Separating compound figures in journal articles to allow for subfigure classification. Proc. SPIE **8674**, 86,740J–86,740J–12 (2013). DOI 10.1117/12.2007897

[45] Chirita, P.A., Firan, C.S., Nejdl, W.: Personalized query expansion for the web. In: Proceedings of the 30th Annual International ACM SIGIR Conference on Research and Development in Information Retrieval, SIGIR '07, pp. 7–14. ACM, New York, NY, USA (2007). DOI 10.1145/1277741.1277746

[46] Choi, S., Lee, J., Choi, J.: SNUMedinfo at ImageCLEF 2013: Medical retrieval task. In: P. Forner, R. Navigli, D. Tufis (eds.) CLEF 2013 Evaluation Labs and Workshop, Online Working Notes (2013). URL `http://ceur-ws.org/Vol-1179/CLEF2013wn-ImageCLEF-ChoiEt2013.pdf`

[47] Cleverdon, C.W.: The significance of the Cranfield tests on index languages. In: Proceedings of the 14th annual international ACM SIGIR conference on Research and development in information retrieval, SIGIR '91, pp. 3–12. ACM, New York, NY, USA (1991). DOI 10.1145/122860.122861

[48] Coletti, M.H., Bleich, H.L.: Medical Subject Headings used to search the biomedical literature. Journal of the American Medical Informatics Association **8**(4), 317–323 (2001). DOI 10.1136/jamia.2001.0080317

[49] Collins-Thompson, K.: Reducing the risk of query expansion via robust constrained optimization. In: Proceedings of the 18th ACM Conference on Information and Knowledge Management, CIKM '09, pp. 837–846. ACM, New York, NY, USA (2009). DOI 10.1145/1645953.1646059

[50] Collins-Thompson, K., Callan, J.: Query expansion using random walk models. In: Proceedings of the 14th ACM International Conference on Information and Knowledge Management, CIKM '05, pp. 704–711. ACM, New York, NY, USA (2005). DOI 10.1145/1099554.1099727

[51] Collins-Thompson, K., Callan, J.: Estimation and use of uncertainty in pseudo-relevance feedback. In: Proceedings of the 30th Annual International ACM SIGIR Conference on Research and Development in Information Retrieval, SIGIR '07, pp. 303–310. ACM, New York, NY, USA (2007). DOI 10.1145/1277741.1277795

[52] Cord, M., Gosselin, P.H.: Image retrieval using long-term semantic learning. In: Image Processing, 2006 IEEE International Conference on, pp. 2909–2912. IEEE (2006)

[53] Crespo, M., Mata, J., Maña, M.: Improving image retrieval effectiveness via query expansion using MeSH hierarchical structure. Journal of the American Medical Informatics Association [online] (2012). DOI 10.1136/amiajnl-2012-000943

[54] Crouch, C.J., Yang, B.: Experiments in automatic statistical thesaurus construction. In: Proceedings of the 15th Annual International ACM SIGIR Conference on Research and Development in Information Retrieval, SIGIR '92, pp. 77–88. ACM, New York, NY, USA (1992). DOI 10.1145/133160.133180

[55] Cummins, R.: Document score distribution models for query performance inference and prediction. ACM Trans. Inf. Syst. **32**(1), 2:1–2:28 (2014). DOI 10.1145/2559170

[56] Cummins, R., Jose, J., O'Riordan, C.: Improved query performance prediction using standard deviation. In: Proceedings of the 34th International ACM SIGIR Conference on Research and Development in Information Retrieval, SIGIR '11, pp. 1089–1090. ACM, New York, NY, USA (2011). DOI 10.1145/2009916.2010063

[57] Datta, R., Joshi, D., Li, J., Wang, J.Z.: Image retrieval: Ideas, influences, and trends of the new age. ACM Comput. Surv. **40**(2), 5:1–5:60 (2008). DOI 10.1145/1348246.1348248

[58] Dempster, A.P., Laird, N.M., Rubin, D.: Maximum likelihood from incomplete data via the EM algorithm. J. Royal Statist. Soc. **39**(1), 1–38 (1977)

[59] Depeursinge, A., Müller, H.: Fusion techniques for combining textual and visual information retrieval. In: H. Müller, P. Clough, T. Deselaers, B. Caputo (eds.) ImageCLEF, *The Information Retrieval Series*, vol. 32, pp. 95–114. Springer Berlin Heidelberg (2010). DOI 10.1007/978-3-642-15181-1_6

[60] Deselaers, T., Keysers, D., Ney, H.: Features for image retrieval: an experimental comparison. Inf. Retr. **11**(2), 77–107 (2008). DOI 10.1007/s10791-007-9039-3

[61] Díaz-Galiano, M.C., Martín-Valdivia, M., Ureña López, L.A.: Query expansion with a medical ontology to improve a multimodal information retrieval system. Comput. Biol. Med. **39**(4), 396–403 (2009). DOI 10.1016/j.compbiomed.2009.01.012

[62] Diethe, T., Hardoon, D.R., Shawe-Taylor, J.: Multiview Fisher discriminant analysis. In: NIPS Workshop on Learning from Multiple Sources (2008)

[63] Dimitrovski, I., Kocev, D., Loskovska, S., Deroski, S.: Hierarchical annotation of medical images. Pattern Recogn. **44**(10-11), 2436–2449 (2011). DOI 10.1016/j.patcog.2011.03.026

[64] Do, M.N., Vetterli, M.: Wavelet-based texture retrieval using generalized Gaussian density and Kullback-Leibler distance. Image Processing, IEEE Transactions on **11**(2), 146–158 (2002)

[65] Doszkocs, T.E.: AID, an associative interactive dictionary for online searching. Online Information Review **2**(2), 163–173 (1978)

[66] Eidenberger, H.: Evaluation and analysis of similarity measures for content-based visual information retrieval. Multimedia Systems **12**(2), 71–87 (2006). DOI 10.1007/s00530-006-0043-z

[67] Eidenberger, H.: Handbook of Multimedia Information Retrieval. Books on Demand, Norderstedt, Germany (2012)

[68] Fan, J., Luo, H., Gao, Y., Jain, R.: Incorporating concept ontology for hierarchical video classification, annotation, and visualization. Trans. Multimedia **9**(5), 939–957 (2007). DOI 10.1109/TMM.2007.900143

[69] Farquhar, J., Hardoon, D., Meng, H., Shawe-Taylor, J.S., Szedmák, S.: Two view learning: SVM-2K, theory and practice. In: Y. Weiss, B. Schölkopf, J.C. Platt (eds.) Advances in Neural Information Processing Systems 18, pp. 355–362. MIT Press (2006)

[70] Fiorini, N., Ranwez, S., Harispe, S., Montmain, J., Ranwez, V.: USI at BioASQ 2015: a semantic similarity-based approach for semantic indexing. In: CLEF 2015 Working Notes, *CEUR Workshop Proceedings, ISSN 1613-0073*, vol. 1391 (2015). URL `http://ceur-ws.org/Vol-1391/94-CR.pdf`

[71] Forner, P., Karlgren, J., Womser-Hacker, C. (eds.): CLEF 2012 Evaluation Labs and Workshop, Online Working Notes (2012). URL `http://clef2012.clef-initiative.eu/index.php?page=Pages/proceedings.php`

[72] Fox, E.A., Shaw, J.A.: Combination of multiple searches. In: D.K. Harman (ed.) NIST Special Publication 500-215: The Second Text REtrieval Conference (TREC-2), p. 23_243. National Institute of Standards and Technology (1994). URL `http://www-nlpir.nist.gov/projects/irlib/pubs/ir/ir_text/ir_text/sp500215_text/23_243.txt`

[73] Gkoufas, Y., Morou, A., Kalamboukis, T.: Combining textual and visual information for image retrieval in the medical domain. Open Medical Informatics Journal **5**, 50–57 (2011)

[74] Gönen, M., Alpaydın, E.: Multiple kernel learning algorithms. J. Mach. Learn. Res. **12**, 2211–2268 (2011)

[75] Gong, Y., Ke, Q., Isard, M., Lazebnik, S.: A multi-view embedding space for modeling internet images, tags, and their semantics. International Journal of Computer Vision **106**(2), 210–233 (2014). DOI 10.1007/s11263-013-0658-4

[76] Gong, Z., Cheang, C., Hou U, L.: Multi-term web query expansion using WordNet. In: S. Bressan, J. Küng, R. Wagner (eds.) Database and Expert Systems Applications, *Lecture Notes in Computer Science*, vol. 4080, pp. 379–388. Springer Berlin Heidelberg (2006). DOI 10.1007/11827405_37

[77] Gonzalo, J., Verdejo, F., Chugur, I., Cigarrän, J.M.: Indexing with WordNet synsets can improve text retrieval. In: Proceedings of the COLING/ACL Workshop on Usage of WordNet in Natural Language Processing Systems, pp. 647–678. Association for Computational Linguistics (1998)

[78] Graupmann, J., Cai, J., Schenkel, R.: Automatic query refinement using mined semantic relations. In: Proceedings of the International Workshop on Challenges in Web Information Retrieval and Integration, WIRI '05, pp. 205–213. IEEE Computer Society, Washington, DC, USA (2005)

[79] Grave, E., Obozinski, G.R., Bach, F.R.: Trace Lasso: a trace norm regularization for correlated designs. In: J. Shawe-Taylor, R.S. Zemel, P.L. Bartlett, F. Pereira, K.Q. Weinberger (eds.) Advances in Neural Information Processing Systems 24, pp. 2187–2195. Curran Associates, Inc. (2011)

[80] Güld, M.O., Thies, C., Fischer, B., Lehmann, T.M.: A generic concept for the implementation of medical image retrieval systems. International Journal of Medical Informatics **76**(2–3), 252 – 259 (2007). DOI 10.1016/j.ijmedinf.2006.02.011

[81] Han, Y., Wu, F., Tao, D., Shao, J., Zhuang, Y., Jiang, J.: Sparse unsupervised dimensionality reduction for multiple view data. IEEE Transactions on Circuits and Systems for Video Technology **22**(10), 1485–1496 (2012). DOI 10.1109/TCSVT.2012.2202075

[82] Harchaoui, Z., Douze, M., Paulin, M., Dudik, M., Malick, J.: Large-scale image classification with trace-norm regularization. In: Computer Vision and Pattern Recognition (CVPR), 2012 IEEE Conference on, pp. 3386–3393 (2012). DOI 10.1109/CVPR.2012.6248078

[83] Hardoon, D.R., Shawe-Taylor, J.: Sparse canonical correlation analysis. Machine Learning **83**(3), 331–353 (2011)

[84] Hastie, T., Tibshirani, R., Friedman, J.: The Elements of Machine Learning, 2nd edn. Springer, New York (2008)

[85] Hastie, T., Tibshirani, R., Friedman, J.: The Elements of Statistical Learning, 2nd edn., chap. 7 (Model Assessment and Selection), pp. 219–260. Springer, New York (2009)

[86] He, J., Li, M., Zhang, H.J., Tong, H., Zhang, C.: Manifold-ranking based image retrieval. In: Proceedings of the 12th annual ACM international conference on Multimedia, pp. 9–16. ACM (2004)

[87] He, R., Tan, T., Wang, L., Zheng, W.S.: $l_{2,1}$ regularized correntropy for robust feature selection. In: Computer Vision and Pattern Recognition (CVPR), 2012 IEEE Conference on, pp. 2504–2511 (2012). DOI 10.1109/CVPR.2012.6247966

[88] García Seco de Herrera, A., Kalpathy-Cramer, J., Demner-Fushman, D., Antani, S., Müller, H.: Overview of the ImageCLEF 2013 medical tasks. In: CLEF 2013 Working Notes, *CEUR Workshop Proceedings, ISSN 1613-0073*, vol. 1179 (2013). URL `http://ceur-ws.org/Vol-1179/CLEF2013wn-ImageCLEF-SecoDeHerreraEt2013b.pdf`

[89] García Seco de Herrera, A., Müller, H.: Fusion techniques in biomedical information retrieval. In: B. Ionescu, J. Benois-Pineau, T. Piatrik, G. Quénot (eds.) Fusion in Computer Vision, Advances in Computer Vision and Pattern Recognition, pp. 209–228. Springer International Publishing (2014). DOI 10.1007/978-3-319-05696-8_9

[90] García Seco de Herrera, A., Müller, H., Bromuri, S.: Overview of the ImageCLEF 2015 medical classification task. In: CLEF 2015 Working Notes, *CEUR Workshop Proceedings, ISSN 1613-0073*, vol. 1391 (2015). URL `http://ceur-ws.org/Vol-1391/172-CR.pdf`

[91] Hersh, W.R.: Information Retrieval: A Health and Biomedical Perspective, 3rd edn. Health informatics. Springer (2009)

[92] Hotelling, H.: Relations between two sets of variates. Biometrika **28**, 321–377 (1936)

[93] Hu, J., Deng, W., Guo, J.: Improving retrieval performance by global analysis. In: Proceedings of the 18th International Conference on Pattern Recognition - Volume 02, ICPR '06, pp. 703–706. IEEE Computer Society, Washington, DC, USA (2006). DOI 10.1109/ICPR.2006.703

[94] Huang, D.A., Wang, Y.C.F.: Coupled dictionary and feature space learning with applications to cross-domain image synthesis and recognition. In: 2013 IEEE International Conference on Computer Vision, pp. 2496–2503 (2013). DOI 10.1109/ICCV.2013.310

[95] Huiskes, M.J., Lew, M.S.: The MIR Flickr retrieval evaluation. In: Proceedings of the 1st ACM International Conference on Multimedia Information Retrieval, MIR '08, pp. 39–43. ACM, New York, NY, USA (2008). DOI 10.1145/1460096.1460104

[96] Hull, D.A.: Stemming algorithms: A case study for detailed evaluation. J. Am. Soc. Inf. Sci. **47**(1), 70–84 (1996). DOI 10.1002/(SICI)1097-4571(199601)47:1<70::AID-ASI7>3.3.CO;2-Q

[97] Iakovidis, D., Pelekis, N., Kotsifakos, E., Kopanakis, I., Karanikas, H., Theodoridis, Y.: A pattern similarity scheme for medical image retrieval. Information Technology in Biomedicine, IEEE Transactions on **13**(4), 442–450 (2009). DOI 10.1109/TITB.2008.923144

[98] Iakovidou, C., Anagnostopoulos, N., Kapoutsis, A., Boutalis, Y., Lux, M., Chatzichristofis, S.: Localizing global descriptors for content-based image retrieval. EURASIP Journal on Advances in Signal Processing **2015**(1), 80 (2015). DOI 10.1186/s13634-015-0262-6

[99] Jégou, H., Douze, M., Schmid, C., Pérez, P.: Aggregating local descriptors into a compact image representation. In: Computer Vision and Pattern Recognition (CVPR), 2010 IEEE Conference on, pp. 3304–3311 (2010). DOI 10.1109/CVPR.2010.5540039

[100] Jelinek, F., Mercer, R.L.: Interpolated estimation of markov source parameters from sparse data. In: E.S. Gelsema, L.N. Kanal (eds.) Proceedings of the Workshop on Pattern Recognition in Practice, Amsterdam, May 21-23, 1980, pp. 381–397. North-Holland (1980)

[101] Jia, Y., Salzmann, M., Darrell, T.: Factorized latent spaces with structured sparsity. In: J.D. Lafferty, C.K.I. Williams, J. Shawe-Taylor, R.S. Zemel, A. Culotta (eds.) Advances in Neural Information Processing Systems 23, pp. 982–990. Curran Associates, Inc. (2010)

[102] Jones, S., Gatford, M., Robertson, S., Hancock-Beaulieu, M., Secker, J., Walker, S.: Interactive thesaurus navigation: Intelligence rules ok? Journal of the American Society for Information Science **46**(1), 52–59 (1995). DOI 10.1002/(SICI)1097-4571(199501)46:1<52::AID-ASI6>3.0.CO;2-1

[103] Jonquet, C., Shah, N.H., Musen, M.A.: The open biomedical annotator. Summit on Translational Bioinformatics **2009**, 56–60 (2009). URL `http://europepmc.org/articles/PMC3041576`

[104] Kalpathy-Cramer, J., de Herrera, A.G.S., Demner-Fushman, D., Antani, S., Bedrick, S., Müller, H.: Evaluating performance of biomedical image retrieval systems—an overview of the medical image retrieval task at ImageCLEF 2004–2013. Computerized Medical Imaging and Graphics **39**(0), 55–61 (2015). DOI 10.1016/j.compmedimag.2014.03.004. Medical visual information analysis and retrieval

[105] Kawamoto, K., Houlihan, C.A., Balas, E.A., Lobach, D.F.: Improving clinical practice using clinical decision support systems: a systematic review of trials to identify features critical to success. British Medical Journal **330**(7494), 765 (2005)

[106] Kennedy, L., Chang, S.F., Natsev, A.: Query-adaptive fusion for multimodal search. Proceedings of the IEEE **96**(4), 567–588 (2008). DOI 10.1109/JPROC.2008.916345

[107] Kosmopoulos, A., Partalas, I., Gaussier, E., Paliouras, G., Androutsopoulos, I.: Evaluation measures for hierarchical classification: a unified view and novel approaches. Data Mining and Knowledge Discovery **29**(3), 820–865 (2015). DOI 10.1007/s10618-014-0382-x

[108] Kou, G., Lu, Y., Peng, Y., Shi, Y.: Evaluation of classification algorithms using MCDM and rank correlation. International Journal of Information Technology & Decision Making **11**(01), 197–225 (2012). DOI 10.1142/S0219622012500095

[109] Krizhevsky, A., Sutskever, I., Hinton, G.E.: ImageNet classification with deep convolutional neural networks. In: F. Pereira, C. Burges, L. Bottou, K. Weinberger (eds.) Advances in Neural Information Processing Systems 25, pp. 1097–1105. Curran Associates, Inc. (2012)

[110] Krovetz, R.: Viewing morphology as an inference process. In: Proceedings of the 16th Annual International ACM SIGIR Conference on Research and Development in Information Retrieval, SIGIR '93, pp. 191–202. ACM, New York, NY, USA (1993). DOI 10.1145/160688.160718

[111] Kulis, B., Darrell, T.: Learning to hash with binary reconstructive embeddings. In: Y. Bengio, D. Schuurmans, J.D. Lafferty, C.K.I. Williams, A. Culotta (eds.) Advances in Neural Information Processing Systems 22, pp. 1042–1050. Curran Associates, Inc. (2009)

[112] Kushki, A., Androutsos, P., Plataniotis, K.N., Venetsanopoulos, A.N.: Query feedback for interactive image retrieval. Circuits and Systems for Video Technology, IEEE Transactions on **14**(5), 644–655 (2004)

[113] Lai, P.L., Fyfe, C.: Kernel and nonlinear canonical correlation analysis. International Journal of Neural Systems **10**(05), 365–377 (2000). DOI 10.1142/S012906570000034X

[114] Lam, W., Ruiz, M., Srinivasan, P.: Automatic text categorization and its application to text retrieval. IEEE Trans. on Knowl. and Data Eng. **11**(6), 865–879 (1999). DOI 10.1109/69.824599

[115] Lam-Adesina, A.M., Jones, G.J.F.: Applying summarization techniques for term selection in relevance feedback. In: Proceedings of the 24th Annual International ACM SIGIR Conference on Research and Development in Information Retrieval, SIGIR '01, pp. 1–9. ACM, New York, NY, USA (2001). DOI 10.1145/383952.383953

[116] Lavrenko, V., Croft, W.B.: Relevance based language models. In: Proceedings of the 24th Annual International ACM SIGIR Conference on Research and Development in Information Retrieval, SIGIR '01, pp. 120–127. ACM, New York, NY, USA (2001). DOI 10.1145/383952.383972

[117] LeCun, Y., Bengio, Y., Hinton, G.: Deep learning. Nature **521**(7553), 436–444 (2015)

[118] Lee, J.A., Verleysen, M.: Nonlinear Dimensionality Reduction. Springer (2007)

[119] Lehmann, T.M., Güld, M.O., Thies, C., Fischer, B., Spitzer, K., Keysers, D., Ney, H., Kohnen, M., Schubert, H., Wein, B.B.: Content-based image retrieval in medical applications. Methods of Information in Medicine **43**(4), 354–361 (2004)

[120] Lehmann, T.M., Schubert, H., Keysers, D., Kohnen, M., Wein, B.B.: The IRMA code for unique classification of medical images. In: Medical Imaging 2003, pp. 440–451. International Society for Optics and Photonics (2003)

[121] Lew, M.S., Sebe, N., Djeraba, C., Jain, R.: Content-based multimedia information retrieval: State of the art and challenges. ACM Trans. Multimedia Comput. Commun. Appl. **2**(1), 1–19 (2006). DOI 10.1145/1126004.1126005

[122] Li, Y., Shi, N., Hsu, D.: Fusion analysis of information retrieval models on biomedical collections. In: Information Fusion (FUSION), 2011 Proceedings of the 14th International Conference on, pp. 1–8 (2011)

[123] Liu, K., Peng, S., Wu, J., Zhai, C., Mamitsuka, H., Zhu, S.: MeSHLabeler: improving the accuracy of large-scale MeSH indexing by integrating diverse evidence. Bioinformatics **31**(12), i339–i347 (2015). DOI 10.1093/bioinformatics/btv237

[124] Liu, S., Liu, F., Yu, C., Meng, W.: An effective approach to document retrieval via utilizing WordNet and recognizing phrases. In: Proceedings of the 27th Annual International ACM SIGIR Conference on Research and Development in Information Retrieval, SIGIR '04, pp. 266–272. ACM, New York, NY, USA (2004). DOI 10.1145/1008992.1009039

[125] Liu, T.Y.: Learning to Rank for Information Retrieval. Springer Berlin Heidelberg (2011). DOI 10.1007/978-3-642-14267-3

[126] Liu, X., Croft, W.B.: Statistical language modeling for information retrieval. Annual Review of Information Science and Technology **39**(1), 1–31 (2005). DOI 10.1002/aris.1440390108

[127] Liu, Y., Li, C., Zhang, P., Xiong, Z.: A query expansion algorithm based on phrases semantic similarity. In: Proceedings of the 2008 International Symposiums on Information Processing, ISIP '08, pp. 31–35. IEEE Computer Society, Washington, DC, USA (2008). DOI 10.1109/ISIP.2008.57

[128] Lowe, H., Barnett, G.: Understanding and using the Medical Subject Headings (MeSH) vocabulary to perform literature searches. JAMA **271**(14), 1103–1108 (1994). DOI 10.1001/jama.1994.03510380059038

[129] Luo, H., Fan, J., Gao, Y., Xu, G.: Multimodal salient objects: General building blocks of semantic video concepts. In: P. Enser, Y. Kompatsiaris, N.E. O'Connor, A.F. Smeaton, A.W. Smeulders (eds.) Image and Video Retrieval, Proc. CIVR, *Lecture Notes in Computer Science*, vol. 3115, pp. 374–383. Springer (2004). DOI 10.1007/978-3-540-27814-6_45

[130] Lux, M., Chatzichristofis, S.A.: LIRE: Lucene Image Retrieval: an extensible Java CBIR library. In: Proceedings of the 16th ACM international conference on Multimedia, MM '08, pp. 1085–1088. ACM, New York, NY, USA (2008). DOI 10.1145/1459359.1459577

[131] Lux, M., Marques, O.: Visual information retrieval using Java and LIRE. Synthesis Lectures on Information Concepts, Retrieval, and Services **5**(1), 1–112 (2013)

[132] Lv, Y., Zhai, C.: Adaptive relevance feedback in information retrieval. In: Proceedings of the 18th ACM Conference on Information and Knowledge Management, CIKM '09, pp. 255–264. ACM, New York, NY, USA (2009). DOI 10.1145/1645953.1645988

[133] Mairal, J., Bach, F., Ponce, J.: Sparse modeling for image and vision processing. Found. Trends. Comput. Graph. Vis. **8**(2-3), 85–283 (2014). DOI 10.1561/0600000058

[134] Mandala, R., Takenobu, T., Hozumi, T.: The use of WordNet in information retrieval. In: Proceedings of the ACL Workshop on the Usage of WordNet in Information Retrieval, pp. 31–37. Association for Computational Linguistics (1998)

[135] Manning, C.D., Raghavan, P., Schütze, H.: Introduction to information retrieval. Cambridge University Press (2008)

[136] Mata, J., Crespo, M., Maña, M.J.: Using MeSH to expand queries in medical image retrieval. In: Proc. MICCAI, Medical Content-Based Retrieval for Clinical Decision Support, MCBR-CDS'11, pp. 36–46. Springer (2012). DOI 10.1007/978-3-642-28460-1_4

[137] Memisevic, R., Sigal, L., Fleet, D.J.: Shared kernel information embedding for discriminative inference. IEEE Transactions on Pattern Analysis and Machine Intelligence **34**(4), 778–790 (2012). DOI 10.1109/TPAMI.2011.154

[138] Metzler, D., Croft, W.B.: Combining the language model and inference network approaches to retrieval. Information Processing & Management **40**(5), 735–750 (2004). DOI 10.1016/j.ipm.2004.05.001. Special Issue on Bayesian Networks and Information Retrieval

[139] Metzler, D., Croft, W.B.: Latent concept expansion using Markov random fields. In: Proceedings of the 30th Annual International ACM SIGIR Conference on Research and Development in Information Retrieval, SIGIR '07, pp. 311–318. ACM, New York, NY, USA (2007). DOI 10.1145/1277741.1277796

[140] Miller, G.A., Beckwith, R., Fellbaum, C., Gross, D., Miller, K.J.: Introduction to WordNet: An on-line lexical database. International Journal of Lexicography **3**(4), 235–244 (1990). DOI 10.1093/ijl/3.4.235

[141] Minker, J., Wilson, G.A., Zimmerman, B.H.: An evaluation of query expansion by the addition of clustered terms for a document retrieval system. Information Storage and Retrieval **8**(6), 329–348 (1972). DOI 10.1016/0020-0271(72)90021-6

[142] Mitchell, T.M.: Machine Learning. McGraw-Hill, New York (1997)

[143] Mitchell, T.M.: Machine Learning, chap. 5 (Evaluating Hypotheses), pp. 128–153. McGraw-Hill, New York (1997)

[144] Müller, H., Clough, P., Deselaers, T., Caputo, B. (eds.): ImageCLEF – Experimental Evaluation in Visual Information Retrieval, *The Information Retrieval Series*, vol. 32. Springer Berlin Heidelberg (2010)

[145] Müller, H., de Herrera, A.G.S., Kalpathy-Cramer, J., Demner-Fushman, D., Antani, S., Eggel, I.: Overview of the ImageCLEF 2012 medical image retrieval and classification tasks. In: Forner et al. [71]. URL `http://www.clef-initiative.eu/documents/71612/ec58b0bf-b68f-423c-abd9-ede306a69cc0`

[146] Müller, H., Michoux, N., Bandon, D., Geissbuhler, A.: A review of content-based image retrieval systems in medical applications – clinical benefits and future directions. International Journal of Medical Informatics **73**(1), 1–23 (2004). DOI 10.1016/j.ijmedinf.2003.11.024

[147] Navigli, R.: Word sense disambiguation: A survey. ACM Comput. Surv. **41**(2), 10:1–10:69 (2009). DOI 10.1145/1459352.1459355

[148] Navigli, R., Velardi, P.: An analysis of ontology-based query expansion strategies. In: Proceedings of the 14th European Conference on Machine Learning, Workshop on Adaptive Text Extraction and Mining, pp. 42–49 (2003)

[149] NCBI, R.C.: Database resources of the National Center for Biotechnology Information. Nucleic Acids Research **41**(D1), D8–D20 (2013). DOI 10.1093/nar/gks1189

[150] NCBI, R.C.: Database resources of the national center for biotechnology information. Nucleic Acids Research **45**(D1), D12–D17 (2017). DOI 10.1093/nar/gkw1071

[151] Nelson, S.J., Johnston, W.D., Humphreys, B.L.: Relationships in Medical Subject Headings (MeSH), pp. 171–184. Springer Netherlands, Dordrecht (2001). DOI 10.1007/978-94-015-9696-1_11

[152] Nie, F., Huang, H., Cai, X., Ding, C.H.: Efficient and robust feature selection via joint $l_{2,1}$-norms minimization. In: J.D. Lafferty, C.K.I. Williams, J. Shawe-Taylor, R.S. Zemel, A. Culotta (eds.) Advances in Neural Information Processing Systems 23, pp. 1813–1821. Curran Associates, Inc. (2010)

[153] Nigam, K., Ghani, R.: Analyzing the effectiveness and applicability of co-training. In: Proceedings of the Ninth International Conference on Information and Knowledge Management, CIKM '00, pp. 86–93. ACM, New York, NY, USA (2000). DOI 10.1145/354756.354805

[154] Porter, M.F.: Readings in Information Retrieval, chap. An Algorithm for Suffix Stripping, pp. 313–316. Morgan Kaufmann Publishers Inc., San Francisco, CA, USA (1997)

[155] Qiu, Y., Frei, H.P.: Concept-based query expansion. In: Proceedings of the 16th Annual International ACM SIGIR Conference on Research and Development in Information Retrieval, SIGIR '93, pp. 160–169. ACM, New York, NY, USA (1993). DOI 10.1145/160688.160713

[156] Quadrianto, N., Lampert, C.H.: Learning multi-view neighborhood preserving projections. In: Proceedings of the 28th International Conference on Machine Learning (ICML'11), pp. 425–432 (2011)

[157] Quellec, G., Lamard, M., Cazuguel, G., Cochener, B., Roux, C.: Wavelet optimization for content-based image retrieval in medical databases. Medical Image Analysis **14**(2), 227–241 (2010). DOI 10.1016/j.media.2009.11.004

[158] Rahman, M., Antani, S., Thoma, G.: A learning-based similarity fusion and filtering approach for biomedical image retrieval using SVM classification and relevance feedback. IEEE Trans. Inf. Tech. Biomedicine **15**(4), 640–646 (2011). DOI 10.1109/TITB.2011.2151258

[159] Rahman, M.M., Desai, B.C., Bhattacharya, P.: Medical image retrieval with probabilistic multi-class support vector machine classifiers and adaptive similarity fusion. Computerized Medical Imaging and Graphics **32**(2), 95 – 108 (2008). DOI 10.1016/j.compmedimag.2007.10.001

[160] Rasiwasia, N., Costa Pereira, J., Coviello, E., Doyle, G., Lanckriet, G.R., Levy, R., Vasconcelos, N.: A new approach to cross-modal multimedia retrieval. In: Proceedings of the 18th ACM International Conference on Multimedia, MM '10, pp. 251–260. ACM, New York, NY, USA (2010). DOI 10.1145/1873951.1873987

[161] Riezler, S., Vasserman, A., Tsochantaridis, I., Mittal, V., Liu, Y.: Statistical machine translation for query expansion in answer retrieval. In: Proceedings of the 45th Annual Meeting of the Association for Computational Linguistics, ACL-07, pp. 464–471. Association for Computational Linguistics (2007)

[162] Rijsbergen, C.J.V.: Information Retrieval, 2nd edn. Butterworth-Heinemann, Newton, MA, USA (1979)

[163] Robertson, S.: Understanding inverse document frequency: On theoretical arguments for IDF. Journal of Documentation **60**(5), 503–520 (2004)

[164] Robertson, S., Zaragoza, H.: The probabilistic relevance framework: BM25 and beyond. Foundations and Trends in Information Retrieval **3**(4), 333–389 (2009). DOI 10.1561/1500000019

[165] Robertson, S.E.: On term selection for query expansion. J. Doc. **46**(4), 359–364 (1990). DOI 10.1108/eb026866

[166] Robertson, S.E., Sparck Jones, K.: Relevance weighting of search terms. Journal of the American Society for Information Science **27**(3), 129–146 (1976). DOI 10.1002/asi.4630270302

[167] Rocchio, J.J.: Relevance feedback in information retrieval. In: G. Salton (ed.) The SMART Retrieval System, pp. 313–323. Prentice-Hall, Englewood Cliffs, NJ (1971)

[168] Rosenfeld, R.: Two decades of statistical language modeling: where do we go from here? Proceedings of the IEEE **88**(8), 1270–1278 (2000). DOI 10.1109/5.880083

[169] Rosipal, R., Krämer, N.: Overview and recent advances in partial least squares. In: C. Saunders, M. Grobelnik, S. Gunn, J. Shawe-Taylor (eds.) Subspace, Latent Structure and Feature Selection: Statistical and Optimization Perspectives Workshop, SLSFS 2005, Revised Selected Papers, pp. 34–51. Springer Berlin Heidelberg (2006). DOI 10.1007/11752790_2

[170] Rota Bulò, S., Rabbi, M., Pelillo, M.: Content-based image retrieval with relevance feedback using random walks. Pattern Recognition **44**(9), 2109–2122 (2011)

[171] Rubin, D.L., Shah, N.H., Noy, N.F.: Biomedical ontologies: a functional perspective. Briefings in Bioinformatics **9**(1), 75–90 (2008). DOI 10.1093/bib/bbm059

[172] Ruthven, I., Lalmas, M.: A survey on the use of relevance feedback for information access systems. The Knowledge Engineering Review **18**(02), 95–145 (2003)

[173] Sackett, D., Rosenberg, W., Gray, J., Haynes, R., Richardson, W.: Evidence based medicine: what it is and what it isn't. British Medical Journal **312**(7023), 71–72 (1996)

[174] Salton, G., Buckley, C.: Term-weighting approaches in automatic text retrieval. Information Processing & Management **24**(5), 513–523 (1988). DOI 10.1016/0306-4573(88)90021-0

[175] Salton, G., Buckley, C.: Improving retrieval performance by relevance feedback. J. Amer. Soc. Info. Science **41**(4), 288–297 (1990). DOI 10.1002/(SICI)1097-4571(199006)41:4<288::AID-ASI8>3.0.CO;2-H

[176] Salton, G., Wong, A., Yang, C.S.: A vector space model for automatic indexing. Commun. ACM **18**(11), 613–620 (1975). DOI 10.1145/361219.361220

[177] Salzmann, M., Ek, C.H., Urtasun, R., Darrell, T.: Factorized orthogonal latent spaces. In: Proc. 13th Int. Conf. Artificial Intelligence and Statistics (AISTATS), pp. 701–708 (2010)

[178] Santosh, K., Xue, Z., Antani, S., Thoma, G.: NLM at ImageCLEF 2015: Biomedical multipanel figure separation. In: CLEF 2015 Working Notes, *CEUR Workshop Proceedings, ISSN 1613-0073*, vol. 1391 (2015). URL `http://ceur-ws.org/Vol-1391/19-CR.pdf`

[179] Saul, L.K., Weinberger, K.Q., Ham, J.H., Sha, F., Lee, D.D.: Semi-Supervised Learning, chap. Spectral Methods for Dimensionality Reduction, pp. 293–308. MIT Press (2006)

[180] Schulz, S., Stenzhorn, H., Boeker, M., Smith, B.: Strengths and limitations of formal ontologies in the biomedical domain. Revista electronica de comunicacao, informacao & inovacao em saude: RECIIS **3**(1), 31–45 (2009). DOI 10.3395/reciis.v3i1.241en

[181] Schütze, H., Pedersen, J.O.: A co-occurrence-based thesaurus and two applications to information retrieval. Inf. Process. Manage. **33**(3), 307–318 (1997). DOI 10.1016/S0306-4573(96)00068-4

[182] Sebastiani, F.: Machine learning in automated text categorization. ACM Comput. Surv. **34**(1), 1–47 (2002). DOI 10.1145/505282.505283

[183] Settles, B.: Active learning literature survey. Computer Sciences Technical Report 1648, University of Wisconsin–Madison (2010)

[184] Shah, N.H., Bhatia, N., Jonquet, C., Rubin, D., Chiang, A.P., Musen, M.A.: Comparison of concept recognizers for building the Open Biomedical Annotator. BMC Bioinformatics **10**(9), S14 (2009). DOI 10.1186/1471-2105-10-S9-S14

[185] Sharma, A., Kumar, A., Daume, H., Jacobs, D.W.: Generalized multiview analysis: A discriminative latent space. In: Computer Vision and Pattern Recognition (CVPR), 2012 IEEE Conference on, pp. 2160–2167 (2012). DOI 10.1109/CVPR.2012.6247923

[186] Shon, A., Grochow, K., Hertzmann, A., Rao, R.P.: Learning shared latent structure for image synthesis and robotic imitation. In: Y. Weiss, B. Schölkopf, J.C. Platt (eds.) Advances in Neural Information Processing Systems 18, pp. 1233–1240. MIT Press (2006)

[187] Shtok, A., Kurland, O., Carmel, D., Raiber, F., Markovits, G.: Predicting query performance by query-drift estimation. ACM Trans. Inf. Syst. **30**(2), 11:1–11:35 (2012). DOI 10.1145/2180868.2180873

[188] Simpson, M.S., Demner-Fushman, D., Antani, S.K., Thoma, G.R.: Multimodal biomedical image indexing and retrieval using descriptive text and global feature mapping. Information Retrieval **17**(3), 229–264 (2014). DOI 10.1007/s10791-013-9235-2

[189] Sindhwani, V., Rosenberg, D.S.: An RKHS for multi-view learning and manifold co-regularization. In: Proceedings of the 25th International Conference on Machine Learning, ICML '08, pp. 976–983. ACM, New York, NY, USA (2008). DOI 10.1145/1390156.1390279

[190] Sivic, J., Zisserman, A.: Video Google: A text retrieval approach to object matching in videos. In: Computer Vision, 2003. Proceedings. Ninth IEEE International Conference on, pp. 1470–1477. IEEE (2003)

[191] Smeulders, A.W.M., Worring, M., Santini, S., Gupta, A., Jain, R.: Content-based image retrieval at the end of the early years. IEEE Trans. Pattern Anal. Mach. Intell. **22**(12), 1349–1380 (2000). DOI 10.1109/34.895972

[192] Smith-Miles, K.A.: Cross-disciplinary perspectives on meta-learning for algorithm selection. ACM Comput. Surv. **41**(1), 6:1–6:25 (2009). DOI 10.1145/1456650.1456656

[193] Snoek, C.G.M., Worring, M.: Concept-based video retrieval. Found. Trends Inf. Retr. **2**(4), 215–322 (2009). DOI 10.1561/1500000014

[194] Sohn, S., Kim, W., Comeau, D.C., Wilbur, W.J.: Optimal training sets for bayesian prediction of MeSH assignment. Journal of the American Medical Informatics Association **15**(4), 546–553 (2008). DOI 10.1197/jamia.M2431

[195] Song, M., Song, I.Y., Hu, X., Allen, R.B.: Integration of association rules and ontologies for semantic query expansion. Data Knowl. Eng. **63**(1), 63–75 (2007). DOI 10.1016/j.datak.2006.10.010

[196] Spall, J.: Multivariate stochastic approximation using a simultaneous perturbation gradient approximation. Automatic Control, IEEE Transactions on **37**(3), 332–341 (1992). DOI 10.1109/9.119632

[197] Spall, J.C.: An overview of the simultaneous perturbation method for efficient optimization. Johns Hopkins APL Technical Digest **19**(4), 482–491 (1998). URL `http://www.jhuapl.edu/techdigest/TD/td1904/spall.pdf`

[198] Sparck Jones, K.: A statistical interpretation of term specificity and its application in retrieval. Journal of Documentation **28**(1), 11–21 (1972)

[199] Stewart, S.A., von Maltzahn, M.E., Sibte, S., Abidi, R.: Comparing MetaMap to MGrep as a tool for mapping free text to formal medical lexicons. In: Proc. KECSM-2012, *CEUR Workshop Proceedings, ISSN 1613-0073*, vol. 895, pp. 63–77 (2012). URL http://ceur-ws.org/Vol-895/paper7.pdf

[200] Sun, R., Ong, C.H., Chua, T.S.: Mining dependency relations for query expansion in passage retrieval. In: Proceedings of the 29th Annual International ACM SIGIR Conference on Research and Development in Information Retrieval, SIGIR '06, pp. 382–389. ACM, New York, NY, USA (2006). DOI 10.1145/1148170.1148237

[201] Sun, S.: Advanced Data Mining and Applications: 7th International Conference, ADMA 2011, chap. Multi-view Laplacian Support Vector Machines, pp. 209–222. Springer, Berlin, Heidelberg (2011). DOI 10.1007/978-3-642-25856-5_16

[202] Sun, S.: A survey of multi-view machine learning. Neural Computing and Applications **23**(7-8), 2031–2038 (2013)

[203] Sun, S., Shawe-Taylor, J.: Sparse semi-supervised learning using conjugate functions. J. Mach. Learn. Res. **11**, 2423–2455 (2010)

[204] Taschwer, M.: Medical case retrieval. PhD Exposé, AAU Klagenfurt (2013). URL http://www.itec.aau.at/~mt/wp/wp-content/uploads/2013/03/expose.pdf

[205] Taschwer, M.: Text-based medical case retrieval using MeSH ontology. In: P. Forner, R. Navigli, D. Tufis (eds.) CLEF 2013 Evaluation Labs and Workshop, Online Working Notes, p. 5. CLEF Initiative, Padua, Italy (2013). URL http://ceur-ws.org/Vol-1179/CLEF2013wn-ImageCLEF-Taschwer2013.pdf

[206] Taschwer, M.: Medical case retrieval. In: Proceedings of the 22nd ACM International Conference on Multimedia, MM '14, pp. 639–642. ACM, New York, NY, USA (2014). DOI 10.1145/2647868.2654856

[207] Taschwer, M.: Textual methods for medical case retrieval. Tech. Rep. TR/ITEC/14/2.01, Institute of Information Technology (ITEC), AAU Klagenfurt, Austria (2014). URL http://www.itec.aau.at/bib/files/textual-mcr.pdf

[208] Taschwer, M., Marques, O.: AAUITEC at ImageCLEF 2015: Compound figure separation. In: CLEF 2015 Working Notes, *CEUR Workshop Proceedings, ISSN 1613-0073*, vol. 1391 (2015). URL http://ceur-ws.org/Vol-1391/25-CR.pdf

[209] Taschwer, M., Marques, O.: Automatic separation of compound figures in scientific articles. Multimedia Tools and Applications pp. 1–30 (2016). DOI 10.1007/s11042-016-4237-x

[210] Taschwer, M., Marques, O.: Compound figure separation combining edge and band separator detection. In: Q. Tian, N. Sebe, G.J. Qi, B. Huet, R. Hong, X. Liu (eds.) MultiMedia Modeling, *Lecture Notes in Computer Science*, vol. 9516, pp. 162–173. Springer International Publishing (2016). DOI 10.1007/978-3-319-27671-7_14

[211] Trieschnigg, D., Pezik, P., Lee, V., de Jong, F., Kraaij, W., Rebholz-Schuhmann, D.: MeSH Up: effective MeSH text classification for improved document retrieval. Bioinformatics **25**(11), 1412–1418 (2009). DOI 10.1093/bioinformatics/btp249

[212] Trieschnigg, R.B.: Proof of concept: Concept-based biomedical information retrieval. Ph.D. thesis, University of Twente, Enschede (2010). DOI 10.3990/1.9789036530644

[213] Tsoumakas, G., Katakis, I.: Multi-label classification: An overview. International Journal of Data Warehousing and Mining (IJDWM) **3**(3), 1–13 (2007)

[214] Tudhope, D., Binding, C., Blocks, D., Cunliffe, D.: Query expansion via conceptual distance in thesaurus indexed collections. Journal of Documentation **62**(4), 509–533 (2006)

[215] Valet, L., Mauris, G., Bolon, P.: A statistical overview of recent literature in information fusion. IEEE Aerospace and Electronic Systems Magazine **16**(3), 7–14 (2001)

[216] Vincent, P., Larochelle, H., Bengio, Y., Manzagol, P.A.: Extracting and composing robust features with denoising autoencoders. In: Proceedings of the 25th International Conference on Machine Learning, ICML '08, pp. 1096–1103. ACM, New York, NY, USA (2008). DOI 10.1145/1390156.1390294

[217] Vinyals, O., Toshev, A., Bengio, S., Erhan, D.: Show and tell: A neural image caption generator. In: The IEEE Conference on Computer Vision and Pattern Recognition (CVPR), pp. 3156–3164 (2015)

[218] Voorhees, E.M.: Query expansion using lexical-semantic relations. In: Proceedings of the 17th Annual International ACM SIGIR Conference on Research and Development in Information Retrieval, SIGIR '94, pp. 61–69. Springer-Verlag New York, Inc., New York, NY, USA (1994)

[219] Voorhees, E.M., Harman, D.K. (eds.): TREC : experiment and evaluation in information retrieval. MIT Press, Cambridge, Mass. (2005)

[220] Wang, K., He, R., Wang, W., Wang, L., Tan, T.: Learning coupled feature spaces for cross-modal matching. In: Proceedings of the 2013 IEEE International Conference on Computer Vision, ICCV '13, pp. 2088–2095. IEEE Computer Society, Washington, DC, USA (2013). DOI 10.1109/ICCV.2013.261

[221] Wang, M., Hua, X.S.: Active learning in multimedia annotation and retrieval: A survey. ACM Trans. Intell. Syst. Technol. **2**(2), 10:1–10:21 (2011). DOI 10.1145/1899412.1899414

[222] Wang, X., Jiang, X., Kolagunda, A., Shatkay, H., Kambhamettu, C.: CIS UDEL working notes on ImageCLEF 2015: Compound figure detection task. In: CLEF 2015 Working Notes, *CEUR Workshop Proceedings, ISSN 1613-0073*, vol. 1391 (2015). URL `http://ceur-ws.org/Vol-1391/65-CR.pdf`

[223] Weiss, Y., Torralba, A., Fergus, R.: Spectral hashing. In: D. Koller, D. Schuurmans, Y. Bengio, L. Bottou (eds.) Advances in Neural Information Processing Systems 21, pp. 1753–1760. Curran Associates, Inc. (2009)

[224] Welling, M., Hinton, G.E.: A new learning algorithm for mean field Boltzmann machines. In: J.R. Dorronsoro (ed.) Artificial Neural Networks — ICANN 2002 Proceedings, pp. 351–357. Springer Berlin Heidelberg (2002). DOI 10.1007/3-540-46084-5_57

[225] White, M., Zhang, X., Schuurmans, D., Yu, Y.l.: Convex multi-view subspace learning. In: F. Pereira, C. Burges, L. Bottou, K. Weinberger (eds.) Advances in Neural Information Processing Systems 25, pp. 1673–1681. Curran Associates, Inc. (2012)

[226] Winkler, F.: Concept detection in biomedical documents. Master's thesis, AAU Klagenfurt (2017)

[227] Wong, W.S., Luk, R.W.P., Leong, H.V., Ho, K.S., Lee, D.L.: Re-examining the effects of adding relevance information in a relevance feedback environment. Inf. Process. Manage. **44**(3), 1086–1116 (2008). DOI 10.1016/j.ipm.2007.12.002

[228] Wu, H.C., Luk, R.W.P., Wong, K.F., Kwok, K.L.: Interpreting TF-IDF term weights as making relevance decisions. ACM Trans. Inf. Syst. **26**(3), 13:1–13:37 (2008). DOI 10.1145/1361684.1361686

[229] Wu, S.: Linear combination of component results in information retrieval. Data Knowl. Eng. **71**(1), 114–126 (2012). DOI 10.1016/j.datak.2011.08.003

[230] Wu, S., Bi, Y., Zeng, X., Han, L.: Assigning appropriate weights for the linear combination data fusion method in information retrieval. Inf. Process. Manage. **45**(4), 413–426 (2009). DOI 10.1016/j.ipm.2009.02.003

[231] Xia, T., Tao, D., Mei, T., Zhang, Y.: Multiview spectral embedding. Trans. Sys. Man Cyber. Part B **40**(6), 1438–1446 (2010). DOI 10.1109/TSMCB.2009.2039566

[232] Xie, B., Mu, Y., Tao, D., Huang, K.: m-SNE: Multiview stochastic neighbor embedding. IEEE Transactions on Systems, Man, and Cybernetics, Part B (Cybernetics) **41**(4), 1088–1096 (2011). DOI 10.1109/TSMCB.2011.2106208

[233] Xu, C., Tao, D., Xu, C.: A survey on multi-view learning. Computing Research Repository **abs/1304.5634** (2013). URL `http://arxiv.org/abs/1304.5634`

[234] Xu, J., Croft, W.B.: Improving the effectiveness of information retrieval with local context analysis. ACM Trans. Inf. Syst. **18**(1), 79–112 (2000). DOI 10.1145/333135.333138

[235] Xu, X., Shimada, A., Taniguchi, R., He, L.: Coupled dictionary learning and feature mapping for cross-modal retrieval. In: 2015 IEEE International Conference on Multimedia and Expo (ICME), pp. 1–6 (2015). DOI 10.1109/ICME.2015.7177396

[236] Xu, X., Yang, Y., Shimada, A., Taniguchi, R.i., He, L.: Semi-supervised coupled dictionary learning for cross-modal retrieval in internet images and texts. In: Proceedings of the 23rd ACM International Conference on Multimedia, MM '15, pp. 847–850. ACM, New York, NY, USA (2015). DOI 10.1145/2733373.2806346

[237] Yang, Y., Nie, F., Xu, D., Luo, J., Zhuang, Y., Pan, Y.: A multimedia retrieval framework based on semi-supervised ranking and relevance feedback. Pattern Analysis and Machine Intelligence, IEEE Transactions on **34**(4), 723–742 (2012)

[238] Ye, T., Wang, T., McGuinness, K., Guo, Y., Gurrin, C.: Learning multiple views with orthogonal denoising autoencoders. In: Q. Tian, N. Sebe, G.J. Qi, B. Huet, R. Hong, X. Liu (eds.) MultiMedia Modeling, *Lecture Notes in Computer Science*, vol. 9516, pp. 313–324. Springer International Publishing (2016). DOI 10.1007/978-3-319-27671-7_26

[239] Yu, J., Wang, M., Tao, D.: Semisupervised multiview distance metric learning for cartoon synthesis. IEEE Transactions on Image Processing **21**(11), 4636–4648 (2012). DOI 10.1109/TIP.2012.2207395

[240] Yu, S., Krishnapuram, B., Rosales, R., Rao, R.B.: Bayesian co-training. J. Mach. Learn. Res. **12**, 2649–2680 (2011)

[241] Yuille, A.L., Rangarajan, A.: The concave-convex procedure. Neural Computation **15**(4), 915–936 (2003). DOI 10.1162/08997660360581958

[242] Zhai, C.: Statistical language models for information retrieval – a critical review. Found. Trends Inf. Retr. **2**(3), 137–213 (2008). DOI 10.1561/1500000008

[243] Zhai, C., Lafferty, J.: Model-based feedback in the language modeling approach to information retrieval. In: Proceedings of the 10th International Conference on Information and Knowledge Management, CIKM '01, pp. 403–410. ACM, New York, NY, USA (2001). DOI 10.1145/502585.502654

[244] Zhai, C., Lafferty, J.: A study of smoothing methods for language models applied to information retrieval. ACM Trans. Inf. Syst. **22**(2), 179–214 (2004). DOI 10.1145/984321.984322

[245] Zhai, D., Chang, H., Shan, S., Chen, X., Gao, W.: Multiview metric learning with global consistency and local smoothness. ACM Trans. Intell. Syst. Technol. **3**(3), 53:1–53:22 (2012). DOI 10.1145/2168752.2168767

[246] Zhang, M.L., Zhou, Z.H.: A review on multi-label learning algorithms. IEEE Transactions on Knowledge and Data Engineering **26**(8), 1819–1837 (2014). DOI 10.1109/TKDE.2013.39

[247] Zhang, N., Man, K.L., Yu, T., Lei, C.U.: Text and content based image retrieval via locality sensitive hashing. Engineering Letters **19**(3), 228–234 (2011)

[248] Zhou, D., Bousquet, O., Lal, T.N., Weston, J., Schölkopf, B.: Learning with local and global consistency. In: S. Thrun, L.K. Saul, B. Schölkopf (eds.) Advances in Neural Information Processing Systems 16, pp. 321–328. MIT Press (2004)

[249] Zhou, X., Depeursinge, A., Müller, H.: Information fusion for combining visual and textual image retrieval. In: Proceedings of the 20th International Conference on Pattern Recognition (2010), ICPR '10, pp. 1590–1593. IEEE Computer Society, Washington, DC, USA (2010). DOI 10.1109/ICPR.2010.393

[250] Zhou, X.S., Huang, T.S.: Relevance feedback in image retrieval: A comprehensive review. Multimedia systems **8**(6), 536–544 (2003)

[251] Zhou, X.S., Zillner, S., Moeller, M., Sintek, M., Zhan, Y., Krishnan, A., Gupta, A.: Semantics and CBIR: a medical imaging perspective. In: Proceedings of the 2008 International Conference on Content-based Image and Video Retrieval, CIVR '08, pp. 571–580. ACM, New York, NY, USA (2008). DOI 10.1145/1386352.1386436

[252] Zhou, Y., Croft, W.B.: Ranking robustness: A novel framework to predict query performance. In: Proceedings of the 15th ACM International Conference on Information and Knowledge Management, CIKM '06, pp. 567–574. ACM, New York, NY, USA (2006). DOI 10.1145/1183614.1183696