Leveraging Document Structures for Analytical and Investigative Information Retrieval

Tuomas Ketola

PhD thesis

School of Electronic Engineering and Computer Science
Queen Mary University of London

2023

Abstract

Due to data leaks, social media and the opening of various government databases, data-driven investigative methods have become available to a wider set of actors, including journalists. Retrieval, search and discovery are vital aspects of these data-driven investigations (DDIs). By contributing to these research areas, this thesis aims to develop methods that can be used in such investigations.

This thesis identifies two characteristics required of retrieval and search methods intended for DDIs: Firstly, the underlying models need to be transparent. Secondly, the models should leverage document structures with and without supervised learning. The core contribution of this thesis is to develop a retrieval method that has these two characteristics and to demonstrate its value in investigative retrieval. By having these characteristics the proposed method — denoted information content field weighting (ICFW) — also contributes to the broader research area of establishing reliable standards for structured document retrieval (SDR).

With respect to ensuring model transparency, the thesis formulates and evaluates formal constraints for SDR. These constraints facilitate the analytical evaluation of existing and proposed models, thus allowing us to reason about their behaviour in a more systematic and logical manner. This adds a layer of transparency to the proposed, as well as existing SDR models, that was not attainable before. In order to leverage the document structure for better performance, ICFW defines the importance given to a document field, not as a semantic property of the collection, but as a statistical property given to each document field. Analysis showing that ICFW satisfies all the proposed constraints for SDR, unlike any existing model, together with a formal evaluation of the method, demonstrates that it is indeed able to leverage document structures in new ways. Finally, the thesis demonstrates that the ICFW method can be used in an investigative retrieval scenario by developing a prototype search system which is evaluated on a hypothetical investigative search task. The system uses the concept of relevance structures to estimate the context in which entities occur in a data collection. These contexts are then used to rank other entities based on the similarity of their context.

Overall, the research presented in this thesis shows that a focus on transparent analytical SDR models has significant potential in advancing investigative retrieval and the field of Information Retrieval in general.

Declaration

I, Tuomas Ketola, confirm that the research included within this thesis is my own work or that where it has been carried out in collaboration with, or supported by others, that this is duly acknowledged below and my contribution indicated. Previously published material is also acknowledged below. I attest that I have exercised reasonable care to ensure that the work is original, and does not to the best of my knowledge break any UK law, infringe any third party's copyright or other Intellectual Property Right, or contain any confidential material. I accept that the College has the right to use plagiarism detection software to check the electronic version of the thesis. I confirm that this thesis has not been previously submitted for the award of a degree by this or any other university. The copyright of this thesis rests with the author and no quotation from it or information derived from it may be published without the prior written consent of the author.

Signature: Tuomas Ketola,

Acknowledgements

I want to express my gratitude to my supervisor Thomas Roelleke who has been immensely helpful and supportive throughout the process. I have learned more about mathematics in our talks than I ever did in school.

My sister Hanna has been the best possible support at every step of the way. Chatting in various pubs and phone calls at the most desperate times is what has kept me going for these years.

Without Dave Moffat this thesis would not exist. In fact, without him I would never have become a computer scientist. Later on, it was our chats at Wetherspoons that guided my research, kept me motivated and got me excited about academia.

Special thanks to Filippo Grassi for taking the time to read through the final draft of my thesis.

Finally, I would like to thank my parents for their support over the years.

This work was funded by the Engineering and Physical Sciences Research Council (EPSRC) as part of the Doctoral Training Centre in Media and Arts Technology at Queen Mary University of London (ref: EP/G03723X/1).

Contents

Glossary

BERT	Bidirectional Encoder Representations from Transformers
BOW	Bag-of-Words
CDF	Cumulative Distribution Function
CNN	Convolutional Neural Network
DFR	Divergence from Randomness (IR model)
FIC	Field Information Content
FRM	Field Relevance Model
FSA	Field Score Aggregation
FSDM	Fielded Sequential Dependence Model
GPT	Generative Pre-trained Transformers,
ICD	Document-based Information Content
ICF	Field-based Information Content
ICFW	Information Content-based Field Weighting
IDF	Inverse Document Frequency
IDJ	Investigative Data Journalism
LSTM	Long Short Term Memory
LTR	Learn-to-Rank
MAP	Mean Average Precision
MLE	Maximum Likelihood Estimate
MLM	Mixture of Language Models
NDCG	Normalized Discounted Cumulative Gain
NLP	Natural Language Processing
PRMS	Probabilistic Relevance Model for Semistructed Data
RNN	Recurrent Neural Network
RSER	Relavance Structure-based Entity Ranking
RSV	Retrieval Status Value
SDR	Stuctured Document Retrieval
SVM	Support Vector Machine
TFA	Term Frequency Aggregation
VSM	Vector Space Model
XML	Extensible Markup Language

Chapter 1

Introduction

1.1 Motivation

The motivation for this PhD thesis can be defined from two perspectives:

1. A technical perspective where the research is motivated by a lack of reliable standard models for analytical structured document retrieval (SDR).
2. A practical perspective where the research is motivated by a lack of methods dedicated to what we define as investigative search, i.e. methods used in data-driven investigations and more specifically in investigative data journalism (IDJ).

The two perspectives are connected, and to an extent follow one another. The technical perspective is contained in the first three content chapters of this thesis, where methods for reliable analytical SDR standards are proposed and analysed. The practical perspective encompasses the start and end points of this thesis: The nature of IDJ dictates that if the proposed models were ever to be useful in the field, they needed to be analytical and structure-focused, which is the starting point for the technical motivations. Furthermore, the thesis finishes with a chapter where a novel investigatory search system is developed and evaluated; a system which was only feasible after reliable standards for analytical SDR had been proposed in the previous chapters. The following will discuss each perspective in detail.

1.1.1 Motivation: Technical Perspective

The majority of data is inherently structured. Whether it is websites, product catalogues, or specific databases, the data has an underlying structure. Probabilistic models, such as the BM25 and LM have become the standard for non-structured (atomic) retrieval, especially if the use of learn-to-rank models is not warranted, or possible. There are many possible reasons why these models have become the standard. In the context of this thesis the following three are considered to be the most important:

1. The models have well-founded conceptual and theoretical models.
2. The models can be shown to rank documents intuitively, meaning in ways that are expected based on widely agreed-upon heuristics.
3. The models have been shown to perform well on a multitude of established test collections.

No such widely accepted standard exists for structured document retrieval (SDR). The fielded extension of the BM25 — the BM25F — could be considered the best candidate. However, it took 20 years for it to become available on commercial systems, such as Elasticsearch and even then it is not the default model. A possible reason for this is that in order to function well, the BM25F requires optimisation, which is not the case for the standard approaches in atomic retrieval. For a retrieval model to be considered a reliable standard — at least to the extent that BM25 and LM are in the context of atomic retrieval — it cannot require optimization. It has to be available "off-the-shelf" and ready to be implemented on any data collection. For an SDR model to accomplish this, it should (in the author's opinion) fulfil the three criteria described above. From a technical perspective the lack of such a model is the underlying motivation for this thesis.

1.1.2 Motivation: Practical Perspective

With growth in the volume, complexity and availability of information, data-driven investigations (DDIs) are becoming more important in many areas. A DDI is any investigation that relies primarily on data. Such investigations are performed in a variety of areas from insurance fraud to law enforcement; from academic research to journalism. What all these investigations have in common is that they are all searching for interesting new facts in the data and that they intend to use these facts in ways that impact other people.

The concept of learning interesting new things from data — rather than simply retrieving information (documents) — relates closely to the field of exploratory search. Exploratory search is any search activity where the user is primarily looking to learn something new from the data. However, in the case of DDIs we need to be more specific: The users are not only looking to learn something new, they are looking for previously unknown facts and furthermore, they are not looking just for themselves, but with the intention of sharing their findings with others. This thesis denotes this type of search as “investigatory search”, closely relating it to “investigative retrieval”.

For exploratory search, the motivation for the searching is for the user to learn something new and interesting, whereas for investigatory search the motivation is to discover facts they can share with others. These facts can of course be interesting, which makes investigatory search a sub-task of exploratory search. The emphasis on facts and other people dictates — to an extent — the kind of methods that can be used with investigatory search systems. When it comes to communicating facts to others the user has to be able to rely on the system not being biased, for example. The methods used have to be transparent to the extent that the user can trust them, meaning they cannot be black-box in nature. This focus on transparency does not need to be as strict for exploratory search in general, since the user might just be looking to learn new things for the fun of it.

The importance of the user being able to communicate the reasoning behind an interesting fact is clear when it comes to using investigatory search systems in the area of investigative data journalism (IDJ). As journalists lack the authority of powerful institutions — unlike law enforcement for example — they have to be able to trust their findings and to back up the facts they are reporting to the rest of society. IDJ makes a good example of DDIs, as the distinction between exploratory and investigatory search is clear. Therefore, IDJ is used as the main example of DDIs in this thesis.

In recent years there have been many examples of IDJ where investigations based on large leaks have led to significant changes in society. One such example is the Panama Papers. Systems have been developed to help journalists scour these databases in these investigations, but little, if any, academic attention has been paid to the subject. From a practical perspective, it is this lack of attention that has motivated this thesis and the technical issues it focuses on.

1.2 Research Objectives and Contributions

The objectives and contributions of this thesis are best described in a chronological manner where the lessons learned from each high-level contribution led to the next. Figure 1.1 and the following description of its components summarise these steps and their specific contributions in detail. Each of the high-level contributions corresponds to a section in the thesis.

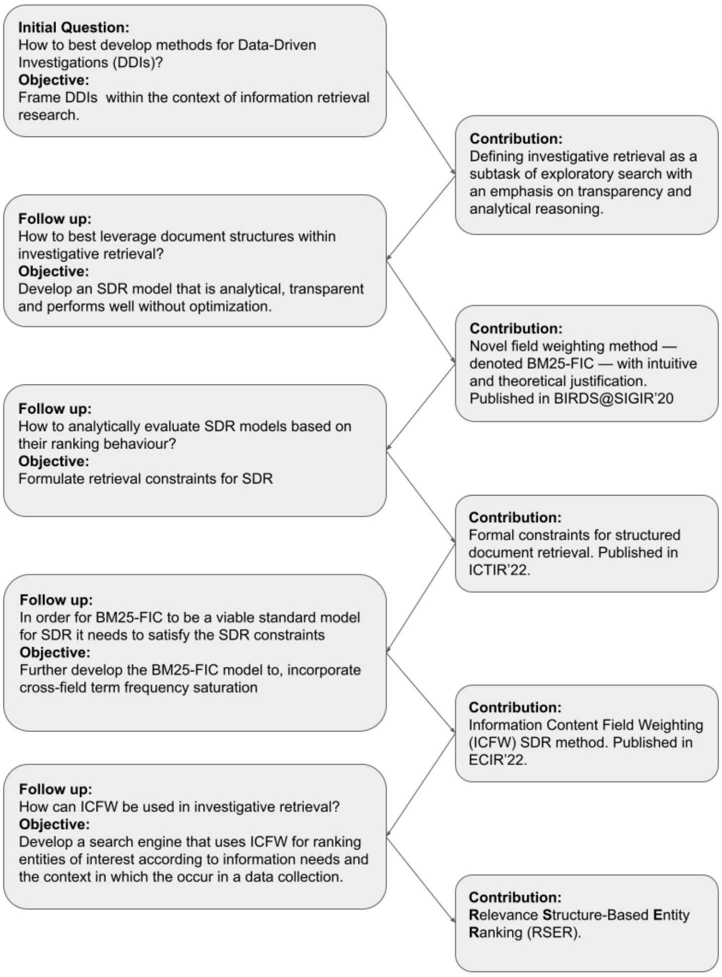

Initial Question:
How to best develop methods for Data-Driven Investigations (DDIs)?
Objective:
Frame DDIs within the context of information retrieval research.
Contribution:
Defining investigative retrieval as a subtask of exploratory search with an emphasis on transparency and analytical reasoning.
Follow up:
How to best leverage document structures within investigative retrieval?
Objective:
Develop an SDR model that is analytical, transparent and performs well without optimization.
Contribution:
Novel field weighting method — denoted BM25-FIC — with intuitive and theoretical justification. Published in BIRDS@SIGIR'20
Follow up:
How to analytically evaluate SDR models based on their ranking behaviour?
Objective:
Formulate retrieval constraints for SDR
Contribution:
Formal constraints for structured document retrieval. Published in ICTIR'22.
Follow up:
In order for BM25-FIC to be a viable standard model for SDR it needs to satisfy the SDR constraints
Objective:
Further develop the BM25-FIC model to, incorporate cross-field term frequency saturation
Contribution:
Information Content Field Weighting (ICFW) SDR method. Published in ECIR'22.
Follow up:
How can ICFW be used in investigative retrieval?
Objective:
Develop a search engine that uses ICFW for ranking entities of interest according to information needs and the context in which the occur in a data collection.
Contribution:
Relevance Structure-Based Entity Ranking (RSER).

Figure 1.1: Thesis Overview

1.2.1 Research Objectives

- Describe the special characteristics and constraints of the retrieval scenario DDIs and IDJ face when reporting from large datasets.

- Develop a retrieval model that fits the characteristics of IDJ by being analytical, dealing well with structure and performing well without optimization.

- Understand how previous SDR models and the proposed model succeed and fail at being a viable standard for analytical SDR.

- Develop an analytical SDR model that fulfils the requirements of becoming a reliable standard for SDR.

- Implement and evaluate a prototype for an investigatory search system.

1.2.2 Research Contributions

The following summarises the core contributions of this thesis. The list has been organized based on the flowchart in Figure 1.1 and the order of the chapters.

- **Discussion: Investigatory search and investigative data journalism.** *Chapter 2*
  - Defining investigatory search by identifying it as a subtask of exploratory search where the user is looking to discover new interesting facts that can be communicated to others.
  - Identifying the need for transparency and capability to leverage document structures as important aspects of investigatory search in the context of IDJ.
- **Information content-based field weighting for SDR.** *Chapter 3*
  - Provide intuitive and theoretical justification for the use of information content for field weighting.
  - Developing BM25-FIC; an SDR method that leverages document structures for increased performance.
  - Formally evaluating the proposed model on benchmark datasets in order to analyse its successes and failures.
- **Formal constraints for SDR.** *Chapter 4*
  - Introducing formal constraints for SDR.
  - Analysis of how existing models and the proposed model from the previous point satisfy and fail to satisfy the constraints.
  - Identifying cross-field term frequency saturation and the consideration of field scores as essential features of a potential reliable standard for SDR.
- **Cross field term frequency saturation in information content-based field weighting.** *Chapter 5*
  - A further iteration of BM25-FIC — denoted information content field weighting (ICFW) — an information-oriented SDR model that incorporates cross-field term frequency saturation and is not specific to BM25, but can be used in place of BM25F.

  - Formally showing how ICFW satisfies the formal constraints for SDR.
  - Evaluating ICFW using well-established benchmarks to demonstrate that it outperforms previous methods and is robust across different structure types.

- **Investigatory search system for ranking entities based on relevance structure.** *Chapter 6*
  - Introducing and implementing the Relevance Structure-Based Entity Ranking (RSER) system, which uses ICFW and user interaction to rank a list of potentially interesting entities based on whether they are found in the data in a context relevant to the user.
  - Building a test collection for the proposed retrieval task.
  - Formally evaluating RSER on the test collection.

1.3 Publications and Submissions

- *BM25-FIC: Information Content-based Field Weighting for BM25F.* Published in BIRDS' workshop at SIGIR 20' [1]. Published (first author)
- *Formal Constraints for Structured Document Retrieval.* Published in ICTIR@SIGIR 22' [2]. Published (first author)
- *Automatic and Analytical Field Weighting for Structured Document Retrieval.* Published in ECIR 23' [3]. Published (first author)
- *Analytical vs Non-Analytical Retrieval: Transparency.* Published in Information Systems 24' (first author) [4].

1.4 Thesis Structure

Chapter 1 — Introduction

Introduces the motivation, research objectives and contributions of the thesis.

Chapter 2 — Motivation and Background

Describes the relevant literature and contextualises how the thesis relates to existing research areas and approaches, especially in terms of Structured Document Retrieval.

Chapter 3 — Information Content-based Field Weighting (ICFW)

Introduces the idea of using information content for field weighting. An intuitive and theoretical justification is provided as well as a study where an initial version of Information Content Field Weighting (ICFW); BM25-FIC is evaluated on two test collections.

Chapter 4 — Formal Constraints for Structured Document Retrieval (SDR)

Presents formal constraints for SDR. The need for analytically evaluating SDR models became apparent in the previous chapter.

Chapter 5 — Term Frequency Saturation in Information Content-based Field Weighting (ICFW)

Details how cross-field term frequency saturation is defined and can be used in ICFW in order for the model to satisfy the SDR constraints from the previous chapter.

Chapter 6 — Relevance Structure-based Entity Ranking and Investigative Information Retrieval (InvIR)

Describes a prototype discovery system for investigative retrieval, where the ICFW model and its field weights are used to define the context in which entities occur in a data collection. This context is used to rank entities of interest according to how well their context matches that of a known entity.

Chapter 7 — Conclusions

Concludes and discusses potential future work.

1.5 Notation

Mathematical notation used throughout the thesis. The notation is repeated at the beginning of appropriate chapters and sections, as well as in text, for extra clarity.

t	a term
d	a document
c	a collection
f	document field: e.g. the title of a document.
F	collection field: e.g. all the titles in the collection.
$n(t,d)$	term frequency: How many times a term t occurs in a document d.
$n(t,f)$	term frequency (in document field f): how many times term t occurs in f.
$n(t,c)$	collection-wide term frequency. How many times term t occurs in collection c.
$\mathrm{TF}_M(t,d,(c))$	term frequency quantification: The term frequency component of a retrieval model M with respect to collection c.
$\mathrm{TF}_M(t,f,(F))$	term frequency quantification: the term frequency component of retrieval model M, with respect to a collection field F
$\mathrm{df}(t,c)$	document frequency: How many documents in collection c have an occurrence of t.
$\mathrm{df}(t,F)$	document frequency (in collection field F): how many times term t occurs in F.
$\mathrm{IDF}(t,c)$	inverse document frequency in collection c.
$\mathrm{IDF}(t,F)$	inverse document frequency in collection field F.
$\mathrm{RSV}_M(q,d,c)$	Retrieval Status Value, i.e. retrieval score for retrieval model M for query q, document d and collection c.
$\mathrm{RSV}_M(q,d,F)$	Retrieval Status Value, i.e. retrieval score for retrieval model M for query q, document d and collection field F.
$N(c)$	the length of the collection, i.e. the total number of documents. As c tends to be implicit, $N(c)$ is usually shorted to N.
$N(F)$	the length of the collection field, i.e. the total number of documents. As F tends to be implicit, $N(F)$ is usually shorted to N.

Chapter 2

Motivation and Background

This chapter begins by discussing investigative data journalism (IDJ) and InvIR, in order to establish a practical grounding to the technical areas described later on in the chapter. This practical grounding affects the areas of IR discussed in the technical part of the background chapter: As the focus of this thesis is on what we call InvIR — where model transparency is instrumental — not much time is spent on black-box learning algorithms for example. Instead, the technical aspects of this chapter concentrate on analytical IR methods designed for both non-structured (atomic) and structured data. Structured data meaning documents that have information in multiple fields and non-structured meaning documents where all the information is in a single field, i.e. there are no fields and no structure. It is the intention of this thesis to transfer lessons learned in atomic retrieval to structured retrieval, both in terms of high-level topics such as retrieval constraints and lower-level technical concerns such as term frequency saturation.

More specifically the chapter is structured as follows:

- Section 2.1 introduces investigative data journalism as an area for IR.
- Section 2.2 establishes the concept of InvIR as a sub-category of exploratory search.
- Section 2.3 discusses the distinction between analytical and non-analytical retrieval models.
- Section 2.4 introduces the relevant non-structured retrieval models.
- Section 2.5 explores the relevant structured models.
- Section 2.6 discusses the importance of term frequency saturation.
- Section 2.7 explores retrieval constraints and axiomatic retrieval.

- Section 2.8 summarises the key aspects of non-structured and structured retrieval models and how they relate to each other in terms of the focus of this thesis.

2.1 Investigative Data Journalism

Before discussing InvIR, it is worth clarifying exactly what is meant by IDJ. Investigative journalism refers to any journalism that has a significant investigative component, as opposed to simply reporting facts that are published by someone else (government, sports organizer etc.) the journalist has to go and do some investigating themselves [5]. Traditionally this has meant finding and talking to sources “on the inside”, confirming their statements through other avenues and reporting the findings. Data journalism is any journalism that has to do with numbers and data. Investigative data journalism means that the story is not reporting solely the numbers and statistics in the data, which is the domain of data journalism. In the last three years everyone has become familiar with a very basic form of data journalism, the daily updates on Covid-19 numbers. Investigative data journalism then not only reports the data, or the numbers but also investigates what is happening behind them. Many of the examples of these investigations have to do with public data and information gained through freedom of information requests. Many important investigations, such as showing the bias of law enforcement towards black people and the horrible state of psychological wards, were done effectively by hand [6, 7].

The focus of this thesis is on investigative data journalism that deals with data sets large enough to warrant the use of automation. A well-known example of such investigations is the tax-heaven leaks that we have seen in recent years. The Panama, Paradise and Pandora papers all reported on millions of documents related to offshore banking, using data leaked from unknown sources. In order to highlight what kind of investigations this thesis is interested in, we will shortly discuss the first of the leaks — the Panama Papers — but all the implications would also apply to their successors as well.

The Panama Papers started with a whistleblower contacting journalists in the Suddeutsche Zeitung in 2015. Throughout the following year or so, the still anonymous source provided more than 11 million files, a data set of 2.6 terabytes. The data was largely unstructured, meaning the database had to be reconstructed before it could be effectively searched and reported on. This took a team of technical experts over a year to accomplish and involved a deal of automation [8]. It has been estimated that by 2021 different countries had recovered 1.3 billion dollars in tax revenue as a direct result of the Panama Papers, meaning the impact of such investigations is a significant one [9].

A set of tools has appeared in the last 10 years to help journalists with exploring the data. The most notable ones are Datashare, developed by the International Consortium for Investigative Journalism (ICIJ) for projects such as the Panama Papers and Aleph developed by the Organized Crime and Corruption Reporting Project (OCCRP) for similar data-driven investigations. At the core of each of these tools are various information retrieval methods. A central argument of this thesis is that these tools and investigations represent a branch of IR that has previously been understudied and warrants special consideration. In this thesis, we denote this branch as Investigative Information Retrieval (InvIR).

2.2 Investigative Information Retrieval (InvIR)

This section introduces the concept of InvIR as a type of exploratory search; or exploratory information retrieval. First, it is worth clarifying some of the notation. Exploratory search refers to any search activity where a user is looking to learn something new from the data, rather than simply looking for factual answers to questions. This means that their information need tends to be much more complicated, meaning their queries tend to be more complex and that often a search session would be comprised of multiple queries [10]. Figure 2.1 by Marchioni [10] shows the different kinds of search tasks that relate to exploratory search. Exploratory information retrieval as a term is not used as widely as exploratory search. In this thesis, the two are used interchangeably.

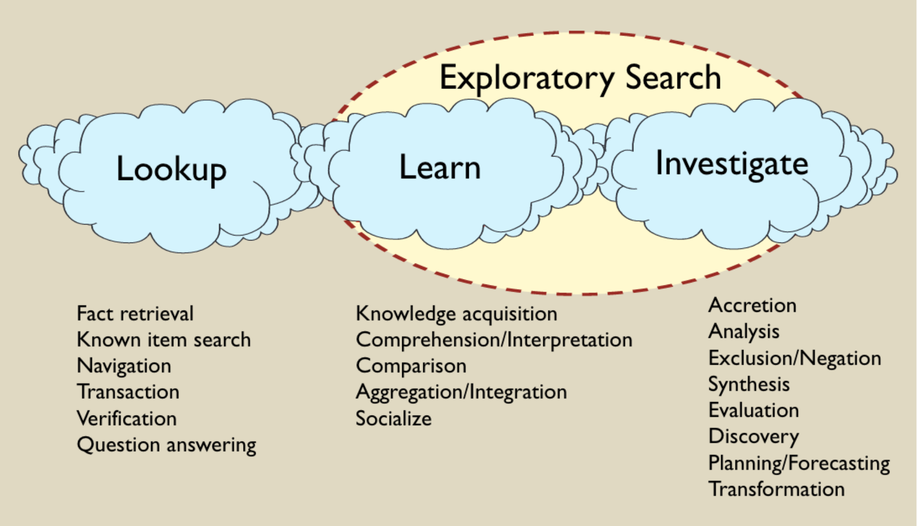


Figure 2.1: Categorization of search activities by Marchioni [10].

InvIR is similar to exploratory IR, in fact, it can be seen as a subset of it.

InvIR refers to any search task where the goal is to learn new facts from the data that are interesting not only to the user but to other people as well. The difference from exploratory search is the emphasis on facts and other people. The IDJ scenario described in the previous section is a clear example of InvIR, as the investigators and journalists are not only searching the data to learn something for themselves but have the aim of reporting their findings to the rest of the world to expose corruption and crime.

The emphasis on other people has implications in terms of what is required of the retrieval system. In order for the information found to be reportable, it has to be believable, meaning the journalist has to understand what they have found, how they have found it and how it can be trusted. This means that an InvIR system has to be transparent and a user has to be able to reason with the system in order to understand all of the relevant information. Table 2.1 illustrates how ad-hoc IR, exploratory IR and InvIR are related. It relates to Figure 2.1 in that search activities shown there can be seen in the light of the three types of IR. Ad-hoc IR deals with more straightforward queries and can therefore be seen as a "lookup" search activity. Exploratory IR covers both learning and investigating search. InvIR can be seen to specifically relate to the rightmost search activities. Table 2.1 looks at the various aspects that make InvIR a sub-category of exploratory search.

Aspect	**Ad-hoc IR**	**Exploratory IR**	**Investigative IR**
Complex Information Needs	Optional	Essential	Essential
Query Reformulation	Optional	Essential	Essential
Session-Based	Optional	Essential	Essential
Complex Results	Optional	Essential	Essential
Complex Data	Optional	Essential	Essential
Transparency	Optional	Optional	Essential
Reasoning	Optional	Optional	Essential

Table 2.1: Differences and similarities between Ad-hoc IR, Exploratory IR and InvIR. The emphasis on transparency and reasoning is what differentiates InvIR from Exploratory IR.

Complex information needs: The complexity of the information need is often reflected in the type of search activity the information need represents. Ad-hoc retrieval essentially covers all possible search activities, meaning for activities such as question answering and verification, the information need is not necessarily complex, whereas for others such as comparison it might be. This is why in Table 2.1 complex information needs is classified as optional.

For exploratory and InvIR information needs tend to be much more complex. Take the comparison search activity from Figure 2.1 for example. The query could be something like "How much more likely is it that it is raining in London, than in Helsinki in June?". Processing such a query often demands more from the system, and in many cases requires the use of entity recognition and multiple queries [10, 11, 12]. Exploratory search and InvIR deal with complex retrieval tasks such as analysis and evaluation, rather than more straightforward ones such as known item search, as demonstrated by Figure 2.1, meaning for a system to be exploratory, or investigative, it must be able to consider complex information needs.

Query Reformulation: as well as complex queries, their reformulation is essential for investigative and exploratory IR. This can be done with help from the system, or completely based on the results the user is seeing [13]. For example, the system can suggest new / better queries, or the user themselves can reformulate their query based on the results they are seeing.

Session-based: the need to handle multiple queries brings forth the need for session-based use of the retrieval system [10, 13]. Unlike in ad-hoc IR, exploratory and InvIR are likely to require multiple queries where each of the queries affects the next. The final information need is then fulfilled by activities during the whole session, most likely involving multiple data collections with intermediate analysis.

Complex Results: with more advanced information needs and a sessions-based approach comes the need for a more complex presentation of the search results. It is likely that a simple ranking of documents with no extra information is not enough for exploratory and InvIR. The user should be given more information regarding each document that has been matched and the underlying reasons for why they have been matched [11, 12].

Transparency: moving to the aspects of the retrieval systems essential for InvIR, but not for all exploratory search activities. The level of transparency required from an investigative system is much higher than other kinds of exploratory search. For example, if a user is simply exploring a dataset in order to learn something new for themselves, it is not necessarily essential that they understand fully how the system has "taught" them. However, underlying the concept of investigations and InvIR is the assumption that the user (investigator) is not only looking for new knowledge for themselves but for others as well. Furthermore, the investigators are not looking to learn just anything, they are looking to learn new facts. IDJ is a clear example of this as the end goal is to have a positive effect on society as a whole using the findings of the investigation. In order for this to be possible, people have to believe the journalists' stories and the facts that they present. If they turn out to be untrue, the journalist's reputation is ruined and the story is dead. For this reason, it is essential from the user's side that they can trust the system and its algorithms, meaning things like black-box deep learning systems with their known and unknown biases are problematic. The next chapter will discuss at length how transparency is defined in the context of this thesis.

Reasoning: it is not only the underlying algorithms of an InvIR system that the user has to trust. Even if the system is not black-box and everything could be traced back to underlying collection statistics, this is not something the user can report to other people and expect to be taken at their (the system's) word. This again is evident from the IDJ example as people would not necessarily trust the word of journalists. Instead, the system has to be able to communicate the reasoning for why it has offered the knowledge it has. Here we are not necessarily talking about an automated system that finds interesting entities in a data collection. The process can be as simple as a user forming connections between entities in the data and reporting on those connections. In such a case the system has to be able to show the user how the connections interact and — to an extent — whether they are valid. Some automation can be involved of course.

The final chapter in this thesis introduces an InvIR application which ranks entities based on how interesting they might be to an investigator based on the context the investigator describes. The key point is that when doing so, the system has to be able to explain its reasoning for the ranking, meaning it has to be able to point to the specific documents in the data, which led to the conclusions. And coming back to the previous point (Transparency) the system also has to explain why those documents are important.

In terms of this thesis, the focus is much more on transparency than reasoning. As the above suggests, the two are closely connected: For example, in terms of transparency, it is beneficial for the system to be able to communicate the reasoning behind the retrieval outcomes. In order to do this, the underlying models must be transparent to an extent at least, otherwise, the system itself cannot fully explain why it has produced the outcome. The point to be made here is that, in many ways transparency is a predecessor of reasoning in the context of InvIR, which is also why it is the main focus of this thesis. Reasoning is only briefly considered in a non-theoretical way in the final chapter.

2.3 Analytical vs Non-Analytical Retrieval

The notion of analytical vs non-analytical retrieval models is key to this entire thesis, so it is worth clarifying it before starting on specific models. This thesis considers a model analytical if its ranking behaviour can be inferred from its specification without further knowledge. For example given a query, two documents and a model (with known hyperparameters), if the model is analytical the ranking of the documents can be inferred, without having to process the documents, or query in any way. A non-analytical model would be one where such inference is not possible. The distinction is important due to the growing popularity of non-analytical learn-to-rank (LTR) models, especially those involving large language models such as BERT [14].

Analytical models are often considered simplistic and not very powerful. As opposed to LTR models, which are non-analytical, they do not leverage training data to the same extent, or in the same manner and therefore usually perform worse on evaluation baselines. However, analytical models have three important advances over non-analytical models:

- They are more transparent.
- They function better if there is no training data.
- They tend to be faster.

The first two points are crucial in the context of my research, as transparency and lack of training data are distinguishing characteristics of an investigative retrieval scenario, which is why the focus in this section and the rest of the thesis is on analytical models.

2.3.1 Analytical vs Non-Analytical Retrieval: Transparency

Transparency in IR, as well as other areas of computer science is a widely debated topic. There is no straightforward, all encompassing definition for it. For the purposes of this thesis, transparency in retrieval is defined using the above distinction between analytical and non-analytical retrieval. Any model where the ranking behaviour can be inferred without having to process the documents or the query, is considered to be transparent. Of course, the transparency of a retrieval model also depends on the observer, i.e. the user. For example, the BM25 as a retrieval model is much more transparent to a user who is familiar with the algorithm, compared to someone who is not. However, the point that is being made here is that no matter who the user is — a model such as the BM25 — where the ranking can be inferred is more transparent than one where it cannot be.

2.4 Non-Structured (Atomic) Document Retrieval

IR is a vast field spanning over 50 years of research and hundreds of different branches. It is not in the scope of this thesis to introduce and consider every single one of them. Instead, we focus on branches and models that are widely used today, introducing their formal definitions, as well as roots in order to provide context. Furthermore, the models we consider must work on any textual data, including cases where no connections between the documents are available. In essence, this means that approaches such as fuzzy retrieval, generalized vector space model, PageRank etc. are not considered here [15, 16, 17, 18].

2.4.1 Notation

It is worth clarifying the notation used in this thesis with regards to terms as there is significant confusion in literature [19].

- $n(t, d)$ = term frequency: How many times a term t occurs in a document d.
- $n(t, c)$ = collection-wide term frequency. How many times term t occurs in collection c. $n(t, c) = \sum_{d \in c} n(t, d)$
- $\mathrm{TF}_M(t, d, (c))$ = term frequency quantification: The term frequency component of a retrieval model M.
- $\mathrm{df}(t, c)$ = document frequency: How many documents in collection c have an occurrence of t.
- $N(c)$ = the length of the collection, i.e. the total number of documents. As c tends to be implicit, $N(c)$ is usually shorted to N.

2.4.2 TF-IDF

Term Frequency - Inverted Document Frequency (TF-IDF) is one of the best-known term weighting methods and perhaps the most popular IR method outside of the field. Originally introduced by Karen Spark-Jones, it gives more emphasis to rare terms making retrieval much more effective [20]. In its original form, the IDF did not have a formal mathematical formulation, but rather an intuitive one: Query terms which appear in many documents are worse at discriminating between them and should therefore be given less weight than those appearing in a few documents [20, 21]. Over the years there have been many interpretations and versions of the TF-IDF model. It is not within the scope of the thesis to summarize all of them. The following will seek to point out the most relevant ones to this thesis.

Spark-Jones introduced the concept of TF-IDF as we know it [20]. The TF-IDF combination represents the concepts of exhaustivity and specificity respectively, where exhaustivity describes how well a term covers a given topic and specificity how well a given term describes a topic. Her important observation was that "It [term specificity] should be interpreted as a statistical rather than semantic property of index terms" [20]. Meaning that collection and term statistics can be used to infer the specificity of terms, rather than having to assign them semantically (manually).

One of the underlying themes of this thesis is to argue that the specificity of a document field, i.e. the weight given to a field of a document, should also be interpreted as a statistical, rather than semantic property.

Inverse Document Frequency

Definition 2.1 (Original Inverse Document Frequency (IDF)). *Let t be a term, c a collection, N the number of documents in c and $\mathrm{df}(t,c)$ the document frequency.*

$$\mathrm{IDF}_{\mathrm{original}}(t,c) := \log \frac{N(c)}{\mathrm{df}(t,c)} \tag{2.1}$$

When describing retrieval model components, the base of the log is generally not relevant as we are dealing with ranking. For this reason, the rest of the thesis only specifies the log when it is relevant.

Definition 2.1 shows the original version of IDF by Spark-Jones [20]. It has no mathematical grounding, which has caused the IDF to have a reputation of being heuristic [19]. However, there have been many attempts to formalise the IDF. One of the most notable and relevant to this thesis is by Robertson et al. They derive what is known as the Robetson-Spark-Jones Weight from the binary independence retrieval (BIR) model [22, 21]. The definitions below demonstrate how the Robetson-Spark-Jones Weight relates to the IDF.

Definition 2.2 (Robetson-Spark-Jones Weight - $w_{\mathrm{rsj,full}}$). *Let r_i be the number of relevant documents containing t in collection c, R the number of relevant documents and r_i the number of relevant documents containing term t.*

$$w_{\mathrm{rsj,full}}(t,c,R,r_i) := \log \frac{(r_i + 0.5)(N - R - \mathrm{df}(t,c) + r_i + 0.5)}{(R - r_i + 0.5)(\mathrm{df}(t,c) + r_i + 0.5)} \tag{2.2}$$

Definition 2.2 assumes relevance knowledge, i.e. knowing R. If this knowledge is not available assuming $R = r_i = 0$ transforms Definition 2.2 to Definition 2.3 [21]:

Definition 2.3 ($w_{\mathrm{rsj,nR}}$ — No relevance knowledge).

$$w_{\mathrm{rsj,nR}}(t,c) := \frac{N - \mathrm{df}(t,c) + 0.5}{\mathrm{df}(t,c) + 0.5} \tag{2.3}$$

Definition 2.3 is very close to the IDF and can be used in its place, however, it becomes problematic for terms that occur in more than half of the documents as the $w_{\text{rsj,nR}}$ value becomes negative. Robertson el al. argue that the $\text{df}(t, c)$ in the numerator can be dropped to fix this issue, thus transforming Definition 2.3 to Definition 2.4 [21, 23].

Definition 2.4 (w_{rsj} — No relevance knowledge / smoothed)**.**

$$w_{\text{rsj,tweak}}(t, c) := \log \frac{N + 0.5}{\text{df}(t, c) + 0.5} \approx \text{IDF}(t, c) \tag{2.4}$$

Definition 2.4 is very close to Definition 2.1 and behaves almost exactly the same way in terms of the "term specificity" values it produces.

Definition 2.5 shows the IDF used by the ElasticSearch library and the Wikipedia article for BM25 at the time of writing[1][2].

Definition 2.5 (IDF-BM25-Elastic-Wiki)**.**

$$\text{IDF}(t, c) := \log \left(\frac{N - \text{df}(t, c) + 0.5}{\text{df}(t, c) + 0.5} + 1 \right) \tag{2.5}$$

Most likely the IDF in Definition 2.5 is an altered version of Robertson et. als RSJ-weight in Definition 2.3. By adding +1 the equation does not produce the unwanted negative values. Why ElasticSearch has not used the method suggested by Robertson et al. themselves (Definition 2.4) is not clear. However, the two of them produce very similar values in real retrieval scenarios.

Definition 2.6 shows another popular IDF variant.

Definition 2.6 (IDF-Sum-Smoothed)**.**

$$\text{IDF}_{\text{sum-smooth}}(t, c) := -\log \frac{\text{df}(t, c) + 0.5}{N + 1} \tag{2.6}$$

Here the smoothing is slightly different with with 0.5 and 1 in the numerator and denominator respectively, compared to 0.5 and 0.5 in other variations.

In Lucene and ElasticSearch (non-BM25 models) the IDF is calculated using Definition 2.7.

Definition 2.7 (IDF-Elastic-Lucene)**.**

$$\text{IDF}_{\text{ES-Luc}}(t, c) := 1 + \log \frac{N + 1}{\text{df}(t, c) + 1} \tag{2.7}$$

Many of the different variations of the IDF can be seen to come out of the different definitions of the document frequency (df). Here we have defined document frequency ($\text{df}(t, c)$) to be the absolute number of documents in collection c

[1] `https://www.elastic.co/blog/practical-bm25-part-2-the-bm25-algorithm-and-its-variables`
[2] `https://en.wikipedia.org/wiki/Okapi_BM25`

in which term t occurs. However, using different notation, as done by [19] we can define different document frequencies.

For the following equations, we use the DF instead of df to keep things clear.

Definition 2.8 (Normalising document frequencies)**.**

$$\mathrm{DF}_{\mathrm{total}}(t,c) := \mathrm{df}(t,c) \tag{2.8}$$

$$\mathrm{DF}_{\mathrm{sum}}(t,c) := \frac{\mathrm{df}(t,c)}{N} \tag{2.9}$$

$$\mathrm{DF}_{\mathrm{sum,smooth}}(t,c) := \frac{\mathrm{df}(t,c)+0.5}{N+1} \tag{2.10}$$

$$\mathrm{DF}_{\mathrm{BIR}}(t,c) := \frac{\mathrm{df}(t,c)}{N-\mathrm{df}(t,c)} \tag{2.11}$$

$$\mathrm{DF}_{\mathrm{BIR,smooth}}(t,c) := \frac{\mathrm{df}(t,c)+0.5}{N-\mathrm{df}(t,c)+0.5} \tag{2.12}$$

The first one does not relate to any of the IDF definitions directly, but rather points out the fact that in reality when we talk of IDF we are actually talking about Inverse normalised Document Frequency. The second one is the basis for Definition 2.1, the third for Definition 2.6, and the third together with the fourth for Definitions 2.3 and 2.4

Going forward when the IDF is discussed in this thesis we are referring to Definition 2.4.

Apart from the BIR-based derivation of the IDF, there have been many attempts to formalize the IDF mathematically. Many of these have to do with information theory and concepts such as entropy, cross-entropy and information content. The relevant attempts will be covered later in the thesis. Outside of information theory approaches worth mentioning include [24, 25].

Term Frequency

In a similar way to the IDF component, the TF component in TF-IDF has seen a lot of change over time. The key issue here is how it is saturated. The original TF was simply calculated as the number of term occurrences in a document.

Definition 2.9 (Raw-TF)**.** *Let $n(t,d)$ be the number of times term t occurs in document d.*

$$\mathrm{TF}_{\mathrm{raw}}(t,d) := n(t,d) \tag{2.13}$$

The issue with this approach is that the first occurrence of term t is given the same importance as a second or third occurrence. This means that the model does not inherently appreciate documents with more unique query terms, something which has been shown to be beneficial by many [26, 27, 28, 2]. In order for the model to do this, term frequency must be saturated. A straightforward and widely used option is to use the LOG-TF.

Definition 2.10 (Log-TF)**.**

$$\mathrm{TF}_{\log}(t, d) := \log\left(1 + \mathrm{TF}_{\mathrm{raw}}(t, d)\right) \tag{2.14}$$

This ensures that the importance given to occurrences of a term decreases as the term frequency increases, i.e. term frequency is saturated. This has been shown to be a useful heuristic by many [26, 27, 29]. Since there is no formal mathematical basis for Equation 2.14 in terms of modelling term dependency it is exactly that; a heuristic.

Vector Space Model

Before moving on to the BM25-model is worth mentioning the vector space model (VSM), which has been widely used together with TF-IDF.

Definition 2.11 (RSV-Vector Space Model)**.** *Let $\overrightarrow{q}$ be a query vector in a space defined by the vocabulary of the collection and $\overrightarrow{d}$ be a document vector defined over the same space.*

$$\mathrm{RSV}_{\mathrm{VSM}}(q, d) := \frac{\overrightarrow{d} \cdot \overrightarrow{q}}{\sqrt{\overrightarrow{d}^2 \cdot \overrightarrow{q}^2}} \tag{2.15}$$

The VSM get its name as the RSV is calculated as the cosine angle between the query and the document. The numerator in Equation (2.15) is the Euclidean norm which helps with the length normalization between the query and document vectors. These vectors can be defined using TF-IDF, which makes the model more effective, than if simple term occurrences are used [19].

Using the cosine "similarity" between vectors is still widely used in IR. For example, advanced deep learning methods still compare query and document vectors, much in the same way as VSM, except they use methods such as word embeddings instead of TF-IDF vectors [30, 31, 32].

2.4.3 Best Match 25 (BM25)

Out of all the retrieval models introduced in this section, the BM25 [33] is by far the most relevant one in terms of this thesis. Even though other models are used in the analysis and experiments as well, the BM25 was the starting point for all the new models introduced in this thesis. The reason for this is that in the last 10 years, it has become the standard in both the academic and commercial space. Furthermore, the way in which it incorporates term frequency saturation has been an important inspiration for the SDR methods proposed in this thesis. The BM25 is sometimes considered an iteration of the TF-IDF model, as its two main components resemble those of the BM25, especially in the absence of relevance knowledge when $w_{\text{rsj,full}} \rightarrow w_{\text{rsj,tweak}}$.

However, this should be considered a misunderstanding of BM25 and TF-IDF. The original iterations of BM models before BM25(15, 11) had four components, one for the RSJ-weight, one for document term frequency, one for query term frequency and one for document length [34]. However, the query term frequency component is usually omitted and term frequency component is combined with the document length component [33]. Robertson el al. developed the BM25 to mirror the probabilistic 2-Poisson model but with a more tangible and simple form [34, 35]. The 2-Poisson aspect of BM25 only concerns the TF component of 2.14, the $w_{\text{rsj,full}}$ is added separately [34].

Definition 2.12 (BM25-TF)**.** *Let k_1 be the term frequency saturation parameter, b the document length normalization parameter and* avgdl(c) *the average document length for collection c.*

$$\text{TF}_{\text{BM25},k_1,b}(t,d,c) := \frac{\text{TF}_{\text{raw}}(t,d)}{\text{TF}_{\text{raw}}(t,d) + K} \tag{2.16}$$

$$K_{k_1,b}(d,c) = k_1 \times \left(1 - b + b\frac{|d|}{\text{avgdl}(c)}\right) \tag{2.17}$$

It is worth re-formulating Equation 2.16 to capture how the document length normalization and term frequency saturation aspects of the BM25 term frequency quantification are connected. Furthermore, this adds clarity later on, as we are largely focused on term frequency saturation, not document length normalization.

$$\text{TF}_{\text{BM25},k_1,b}(t,d,c) := \frac{\text{TF}_{\text{piv,b}}(t,d,c)}{\text{TF}_{\text{piv,b}}(t,d,c) + k_1} \tag{2.18}$$

where $\text{TF}_{\text{piv,b}}$ is the document length normalized term frequency:

Definition 2.13.

$$\text{TF}_{\text{piv,b}}(t,d,b,c) := \frac{\text{TF}_{\text{raw}}(t,d)}{b \times \frac{|d|}{\text{avgdl}(c)} + (1-b)} \tag{2.19}$$

Definition 2.14 (RSV BM25)**.**

$$\text{RSV}_{\text{BM25},k_1,b}(d,q,c,R) := \sum_{t \in q} \text{TF}_{\text{BM25}}(t,d,c) \cdot w_{\text{rsj,full}}(t,c,R) \tag{2.20}$$

The hyperparameters in BM25-TF are k_1, which determines the scale of term frequency saturation and b which determines the scale of document length normalization. Term frequency saturation is one of the concepts in this thesis and therefore it will be discussed at length in a separate section (Section 2.6). The other hyperparameter is b, which controls the degree of document length normalization. Figure 2.2 demonstrates the effect of b on BM25-TF.

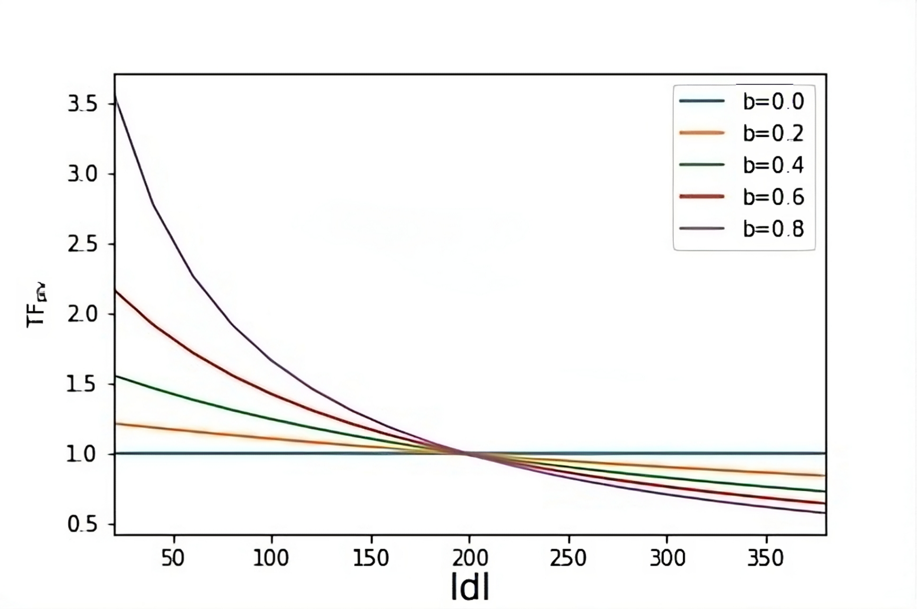


Figure 2.2: The effect of document length normalization on term frequency quantification for BM25 with different values of b.

Even though less emphasis is given to document length normalization, compared to term frequency saturation in this thesis, it is worth noting that it is an important factor in making BM25 as powerful as it is. It is also worth noting that the widely accepted good ranges for b (0.75-0.80) and k_1 (1.2-2.0) have made it possible for BM25 to become popular, which as we will see later, is a problem when it comes to SDR as these ranges do not seem to apply. Closely related to the BM25-TF is Paiks-TF, where term burstiness is considered as well [36].

2.4.4 Language Modelling (LM)

LM became popular in the late 90s and has since been considered one of the main benchmark models in IR research. This section introduces and clarifies the theory underlying LM.

LM is based on a different conceptual model compared to the BM25. Where BM25 seeks to model the probability of a document being relevant given a query, LM instead estimates the probability of a query given a document: $P(q|d)$ rather than $P(d = \text{relevant}\,|q)$. This distinction between $P(q|d)$ and $P(d = \text{relevant}\,|q)$ has been discussed at length by Roelleke [19].

Intuitively, LM estimates the probability of producing a query by randomly choosing words from a document. Problems with this conceptual model arise if there are words in the query that are not found in the document. If the ranking score for a document was calculated as the product of the query term probabilities within a document, all documents that do not contain every single query term would get a score of 0. This of course is not desirable, since those documents could still be useful. In order to include these documents in the retrieval results, LM not only considers the probability of terms given a document $P(t|d)$, but also the probability of terms given a collection as a whole $P(t|c)$. The two probabilities are mixed in different ways and are commonly referred to, as the foreground and background models respectively.

Definition 2.15 (RSV-LM)**.** *Let λ be the mixture parameter, determining the weight given to the foreground and background models.*

$$\mathrm{RSV}_{\mathrm{LM}}(d,q,c) := \sum_{t \in q} \mathrm{TF}_{\mathrm{raw}}(t,q) \cdot \log\left[(1-\lambda)P(t|d) + \lambda P(t|c)\right] \qquad (2.21)$$

It is worth noting that in literature sometimes the mixture parameter is defined the other way around as: $\mathrm{RSV}_{\mathrm{LM}}(d,q,c) := \sum_{t \in q} \mathrm{TF}_{\mathrm{raw}}(t,q) \cdot \log(\lambda P(t|d) + (1-\lambda)P(t|c))$ [19]. In this thesis Definition 2.15 is preferred as it is used by the Lucene system and thus makes the experimentation later on more straightforward.

By normalizing Definition 2.15 and transforming using methods described by Roelleke [19] we can derive the following definition for RSV-LM.

Definition 2.16 (RSV-LM-Normalized)**.**

$$\mathrm{RSV}_{\mathrm{LM,norm}}(d,q,c) := \sum_{t \in q} n(t,q) \cdot \log\left(1 + \frac{(1-\lambda)}{\lambda}\frac{P(t|d)}{P(t|c)}\right) \qquad (2.22)$$

In most cases $P(t|d)$ and $P(t|c)$ are defined as the maximum likelihood estimate of the probability of term t under the term distribution of for document d, or collection c as defined below [37]:

Definition 2.17 (LM Term Probabilities)**.** *Let $n(t,c)$ be the collection-wide* *term frequency, i.e. the number of occurrences of term t in all the documents.*

$$P(t|d) = \frac{n(t,d)}{\sum_{t_i \in d} n(t_i, d)} \tag{2.23}$$

$$P(t|c) = \frac{n(t,c)}{\sum_{t_i \in c} n(t_i, c)} \tag{2.24}$$

The best-known variations of LM models are about estimating the value for λ, as it determines the trade-off between the background model and the foreground model. These methods are often called smoothing methods, as they smooth the effect of the background model on the RSV.

The most straightforward smoothing method is the Jelenik-Mercer method. There λ is set as a constant between 0 and one. Widely used retrieval libraries such as Elastic Search and Lucene set $\lambda = 0.1$, though they mention this works better for short queries, rather than long ones[3]. A more complex smoothing method is Dirichlet-based smoothing.

Definition 2.18 (LM - Dirichlet-based smoothing)**.**

$$\lambda := \frac{|d|}{\mu + |d|} \tag{2.25}$$

A case can be made to set μ as a function of avgdl [19]. In general values between 200 and 2000 have been recommended [38].

A more recent model by Cummings el al. introduces the smoothened Polya urn document (SPUD) language model where the multinominal distribution, used in most language models, is replaced with the Dirichlet compound multinominal [29]. This has a strong effect on how term frequency saturation is modelled, as we will see in Section 2.6.

2.4.5 Divergence from Randomness (DFR)

Amati el al. introduced the DFR model in the early 2000s [39]. It represents a third conceptual model for how documents are to be retrieved that we discuss in this section, the first one being the probability of relevance (BM25) and the second one query likelihood (Language Modelling). The underlying idea in DFR is to score documents based on how divergent they are from being random. Conceptually, if we have a matching term with a high likelihood to appear in a document — such as the word "the" — the model will not give it a lot of weight, even if it occurs many times. The DFR model is covered at some length here as

[3] https://www.elastic.co/guide/en/elasticsearch/reference/current/index-modules-similarity.html

it is based around information content, the same as the proposed models later in this thesis.

At the core of the DFR model lies the two probability mass functions described below. The significance of defining them as probability mass functions, rather than cumulative probability functions, or density functions is discussed later on in the section.

Definition 2.19 (DFR Probability 1 — Prob1). *Let Θ be a model of randomness. In Amati et al. [39] Θ = [Bernoulli-Poisson, Bernoulli-Divergence, Bose-Einsten, TF-IDF]*

$$\mathrm{Prob1}(t, d, c, \Theta) := P(n(t, d)|\Theta) \tag{2.26}$$

Definition 2.20 (DFR Probability 2 — Prob2). *Let ϑ be a model of randomness. In [39] $\vartheta \in$ [Laplace, Bernoulli]*

$$\mathrm{Prob2}(t, d, \vartheta) := P(n(t, d)|\vartheta) \tag{2.27}$$

Definition 2.21 (DFR Informative Content 1 — Inf1).

$$\mathrm{Inf1}(t, d, c, \Theta) := -\log_2 \mathrm{Prob1}(t, d, \Theta) \tag{2.28}$$

Definition 2.22 (DFR Informative Content 2 — Inf2).

$$\mathrm{Inf2}(t, d, \vartheta) := 1 - \mathrm{Prob2}(t, d, \vartheta) \tag{2.29}$$

Definition 2.23 (Divergence from Randomness RSV).

$$\mathrm{RSV}_{\mathrm{DFR}}(d, q, c) := \sum_{t \in q \cap d} [\mathrm{TF}_{\mathrm{raw}}(t, q) \cdot \mathrm{Inf1}(t, d, \Theta) \cdot \mathrm{Inf2}(t, d, c, \vartheta)] \tag{2.30}$$

As is evident from Definition 2.23 the weight given to a term depends on two kinds of informative content. Here it is important to distinguish between information content as it is understood in information theory and as a component of entropy and informative / information content as it is understood in terms of what Hintikka [40] calls semantic information theory. DFR is based on the latter. According to Hintikka's definition, both Definition 2.21 and 2.22 are valid measures of informative content. Amati el al. provide formal proof for why the two informative content measures are combined in Definition 2.23, by considering Inf2 as a normalizing component which discounts the effect of Inf1 based on the concept of risk, which they borrow from utility theory.

The DFR model is comprised of three components: The basic model, the first normalization and the second (document length) normalization. Here the focus will be on the first two, as document length normalization is not a core aspect of this thesis.

Basic Models of Randomness

The basic model calculates the probability and therefore the information content of a document having a given number of query term occurrences. The models of randomness considered by Amati et al. [39] are the Bernoulli model, the Bose-Einstein model, the tf-idf model, the tf-itf model and the tf-expected_idf model. The Bernoulli model is approximated using the Poisson model and a divergence model. The Bose-Einstein model is considered using two limiting formulae the geometric distribution and using Stirling's formulae.

Not all these options are discussed at length here. Three of the above basic models are discussed in detail for the following reasons: the Poisson-based approximation of the Bernoulli model as it relates to the formulation of the BM25 (P in [39]), the Bose-Einstein model and the TF-IDF model (I(n) in [39]).

For the Bernoulli model of randomness $\mathrm{Inf1}_{\mathrm{Bern}}$ is defined as:

Definition 2.24 (Bernoulli Model of Randomness)**.** *Let* $p = \frac{1}{N}$ *and* Bern *be a model of randomness based on the Bernoulli distribution.*

$$\mathrm{Prob1}(t, d, c, \mathrm{Bern}) := \binom{n(t,c)}{n(t,d)} p^{n(t,d)} (1-p)^{(n(t,c)-n(t,d))} \tag{2.31}$$

Slightly confusingly [39] et al. call this model of randomness Bernoulli, even though it represents the binomial distribution.

Following Definition 2.21 for Inf1 the expanded expression is as follows:

$$\mathrm{Inf1}(t, d, c, \mathrm{Bern}) = -\log\left[\mathrm{Prob1}(t, d, c, \mathrm{Bern})\right] \tag{2.32}$$

Assuming that p decreases towards 0 as N increases, the formula above can be approximated using the Poisson distribution:

$$\mathrm{Inf1}(t, d, c, \mathrm{Bern}) \approx -\log \mathrm{Pois}_{\lambda}(n(t,d)) \tag{2.33}$$

$$\mathrm{Pois}_{\lambda}(n(t,d)) = \frac{e^{-\lambda}\lambda^{n(t,d)}}{n(t,d)!} \tag{2.34}$$

where $\lambda = p \times n(t,c)$. Setting λ in this way is important as λ defines the expected probability of randomly observing $n(t,d)$ occurrences of term t in document d. By setting $p = \frac{1}{N}$, the model guarantees that there are no cases where the $P(n(t,d)) > P(n(t,d)+1)$, as $\frac{1}{N}$ represents the minimum probability we can observe for a term.

The Bernoulli model of randomness is as the Binominal distribution, where a single Bernoulli trial models the probability of a term occurring, or not occurring in a document.

The second basic model considered here is the Bose-Einstein model. Suppose that we randomly place $n(t,c)$ tokens of a term in N documents. This "event" completely describes its occupancy numbers: $n(t,d_1),\ldots,n(t,d_N)$. For all the possible events the following is true:

$$n(t,d_1)+\cdots+n(t,d_N)=n(t,c) \tag{2.35}$$

The number of possible configurations for Equation 2.35 is defined as s_1 and is given by the binomial coefficient:

$$s_1=\binom{N+n(t,c)-1}{n(t,c)}=\frac{(N+n(t,c)-1)!}{(N-1)!n(t,c)!} \tag{2.36}$$

s_1 denotes the total number of combinations possible given N documents and $n(t,c)$ term occurrences. In order to calculate the probability of document d_k having exactly $n(t,d)$ occurrences of term t, the number of configurations that produce such an outcome must also be considered. This number is denoted as s_2 can be calculated by considering Equation 2.36 without the document k:

$$\begin{split}n(t,d_1)+\cdots+n(t,d_{k-1})+\\ \cdots+n(t,d_{k+1})+\cdots+n(t,d_N)=n(t,c)-n(t,d_k)\end{split} \tag{2.37}$$

In a similar manner to s_1 the number of possible combinations for this event can be calculated as:

$$\begin{split}s_2&=N-1+(n(t,c)+n(t,d_k)-1)n(t,c)-n(t,d_k)\\ &=\frac{(N+n(t,c)-n(t,d_k)-2)!}{(N-2)!(n(t,c)-n(t,d_k))!}\end{split} \tag{2.38}$$

Assuming that $n(t,c)\gg 1$, $N\gg 1$ and that $\frac{n(t,c)}{N}=O(1)$ (same as for Poisson above), both s_1 and s_2 follow the Bose-Einstein distribution. The probability of document d_k having exactly $n(t,d)$ occurrences of term t can then be calculated in using combinations, with s_1 representing all possible combinations given a collection c and s_2 all possible combinations given c, without document k. For the Bose-Einstein model of randomness Prob1 is calculated as:

Definition 2.25 (Bose-Einstein Model of Randomness)**.**

$$\text{Prob1}(t,d,c,\text{Bose-Ein})=\frac{s_2}{s_1} \tag{2.39}$$

It is worth noting that the Bose-Einstein model of randomness is not really a probability model based on a Bose-Einstein "probability" distribution, but a probability calculated using combinations which are calculated using binomial coefficients. These binomial coefficients follow the Bose-Einstein distribution due to the above-mentioned assumptions about $n(t, c)$, N and $\frac{n(t,c)}{N}$.

Equations (2.36) and (2.38) are related to the derivation of the Bose-Einstein distribution from the microcanonical ensembles. Another potential derivation of the Bose-Einstein distribution comes from the grand canonical ensemble. Part of this derivation states that the probability distribution for the number of bosons (term occurrences in IR) is a geometric distribution. The details underlying the derivation of the Bose-Einstein distribution using microcanonical ensembles and the grand ensemble venture too far into the world of physics for us to spend time on it here.

Following Definition 2.21 for Inf1 the expanded expression for Equation 2.39 is as follows:

$$\text{Inf1}(t, d, c, \text{Bose-Ein}) = -\log\left[\text{Prob }1(t, d, c, \text{Bose-Ein})\right] \tag{2.40}$$

The third basic model considered here is the TF-IDF randomness model. It is discussed here at length as the IDF is a concept that is prevalent in many parts of this thesis, both in terms of implementation and inspiration. The TF-IDF randomness model computes Prob1 by first computing the unknown probability of p of choosing a document at random and then computing the probability of having $n(t, d)$ occurrences of t in that document. Using a Bayesian approach and various assumptions about the underlying distributions, as well as the distribution features, [39] arrive at the following definition for the probability of randomly choosing a document with the term t:

$$P(t \in d|c) := \frac{\text{df}(t, c) + 0.5}{N_D(c) + 1} \tag{2.41}$$

They further assume that the occurrence of a term t is independent of all other term occurrences including those of term t, meaning the probability of observing $n(t, d)$ occurrences of term t is defined as:

Definition 2.26 (TF-IDF Model of Randomness)**.**

$$\text{Prob1}(t, d, c, \text{TF-IDF}) := P(t \in d|c)^{n(t,d)} \tag{2.42}$$

Following Definition 2.21 for Inf1 the expanded expression is as follows:

$$\begin{aligned}
&\text{Inf1}(t, d, c, \text{TF-IDF}) \\
&\qquad = n(t, d) \cdot \log \frac{N + 1}{\text{df}(t, c) + 0.5} \propto \text{TF}_{\text{raw}}(t, d) \cdot \text{IDF}(t, c) \qquad (2.43)
\end{aligned}$$

From Equation 2.43 it is evident that the TF-IDF model of randomness closely resembles the TF-IDF model with the raw term frequency quantification, assuming an $\log \frac{N+1}{\text{df}(t,c)+0.5}$ like IDF. This means that term frequency is not saturated by Equation 2.43.

First Normalization

The underlying idea behind the first normalization (Inf2) is to model what [39] calls the after-effect of sampling. That is, Prob2 is focused only on modelling the probabilities of term occurrences in documents that have an occurrence of the term (the elite set). It is assumed that the probability Prob2$(n(t, d))$ is obtained by the conditional probability $P(n(t, d)+1|n(t, d))$, i.e. the probability of having one more occurrence of t in the document. The probability $P(n(t, d)+1|n(t, d))$ is obtained using the aftereffect model. [39] introduce two aftereffect models, one based on the Laplace law of succession and one on Bernoulli trials and an urn model. Here we concentrate on the former, which shares similarities with the BM25 term frequency quantification (TF_{BM25}). The Laplace model of aftereffect calculates Prob2 as

$$\text{Prob2}(t, d) = P(n(t, d) + 1|n(t, d)) = \frac{n(t, d)}{n(t, d) + 1} \qquad (2.44)$$

Following Definition 2.22:

Definition 2.27 (Laplace Aftereffect model)**.**

$$\text{Prob2}(t, d, \text{Laplace}) := \frac{n(t, d)}{n(t, d) + 1} \qquad (2.45)$$

$$\text{Inf2}(t, d, \text{Laplace}) = \frac{1}{1 + n(t, d)} \qquad (2.46)$$

From Equation 2.27, as $n(t, d)$ grows Inf2 decreases. Together with the basic randomness model, this means that term frequency is saturated. For example, even though the TF-IDF basic model of randomness ($\text{Inf1}_{\text{TF-IDF}}$) uses the raw term frequency after it is multiplied by Inf2 term frequency is saturated. The nature of this saturation is similar to TF_{BM25}, which is obvious from Equation 2.45 (with $k_1 = 1$).

The TF-IDF basic model with Laplace first normalization is used in the experimentation of this thesis as it shares many similarities with the proposed models: Firstly, they both are based on information content, secondly, the underlying inspiration is related to the IDF and finally, the term frequency saturation of Laplace smoothing is similar to that of the BM25, which is one of the underlying motivations for the proposed models in this thesis.

The Underlying Purpose of Inf1 and Inf2

As discussed earlier there are two important aspects that all the atomic retrieval models discussed in this chapter must possess in order for them to function well: 1. They should emphasise terms that are better discriminators, i.e. they are rare and 2. Term Frequency should be saturated. The DFR model — much like the BM25 — splits these two model attributes into two components. Inf1 makes sure the rare terms receive more emphasis and Inf2 (or first normalization) saturates the effect of higher term frequencies.

More formally, consider the Bernoulli-Poisson-based Prob1. Definition 2.24 represents a probability mass function (PMF) for the number of occurrences of term t being equal to $n(t, d)$ $(P(n(t, d)|d, c)))$. What makes Prob1 and Prob2 probability mass functions, rather than probability density functions is their inherent discrete nature. The DFR model could also be defined not through the probability mass function, but the cumulative distribution function (CDF) $(P(n(t, d) <= k_t)$ where k_t is the number of times term t is observed in a document). This approach has been denoted as first-generation DFR (DFR-1) in literature [19]. Amati et al. only focus on the mass function-based approach (DFR-2 in [19]), which is also what we have done here. As the PMF is calculated based on the number of occurrences of term t in a document $(n(t, d))$ and the number of occurrences in a collection $(n(t, c))$, the higher the value for $n(t, c)$ is, the higher the value for Prob1 is, meaning the higher $n(t, c)$ is the lower Inf1 is. This is the underlying purpose of Inf1; to emphasise rare terms. As mentioned earlier, the purpose of Inf2 is to saturate term frequency.

Notes on Implementations

In terms of commercial implementation, different libraries offer different options for the DFR Basic models and the second normalization. Lucene and libraries built on top of it such as Elasticsearch have the Geometric Bose-Einstein model and the three different TF-IDF models. For the aftereffect both the Laplace and Bernoulli options are available. The same goes for the SOLR library.

Thesis Baseline DFR Model

The experimentation in Chapter 5 uses DFR with a TF-IDF basic model, Laplace as the first normalization and H1 as the second normalization.

Definition 2.28 (RSV DFR Baseline)**.** *Let* $n_{\text{norm}}(t,d) = n(t,d) \cdot \frac{\text{avgdl}}{|d|}$*, i.e. the length normalized term frequency.*

$$\begin{aligned}
&\text{RSV}_{\text{DFR,BL}}(d, q, c, \text{TF-IDF}, \text{Laplace}) \\
&\quad := \sum_{t \in q \cap d} \text{Inf1}(t, d, c, \text{TF-IDF}) \cdot \text{Inf2}(t, d, \text{Laplace}) \\
&\quad = \sum_{t \in q \cap d} n_{\text{norm}}(t,d) \cdot \log \frac{N+1}{\text{df}(t,c) + 0.5} \cdot \frac{1}{1 + n_{\text{norm}}(t,d)} \qquad (2.47)
\end{aligned}$$

This configuration was chosen for the following reasons: 1. It is shown to perform relatively well by [39], 2. It uses the TF-IDF as an underlying model, a feature shared with our proposed model and 3. The Laplace first normalization shares similarities with the BM25, which has had a large influence on models presented in this thesis.

The last point was also made by Amati [39] in their original paper. The following provides a simplified explanation for clarity in the context of this thesis. Rearranging Equation 2.47 we have:

$$\text{RSV}_{\text{DFR,BL}}(d, q, c, \text{TF-IDF}, \text{Laplace}) = \frac{n_{\text{norm}}(t,d)}{1 + n_{\text{norm}}(t,d)} \cdot \log \frac{N+1}{\text{df}(t,c) + 0.5} \qquad (2.48)$$

If we assume that the second term corresponds to the IDF (more specifically, Definition 2.6), no document length normalization and that the k_1 value for BM25 is equal to 1, the ranking functions (Equations 2.47 and Equation 2.20) are the same.

An argument could be made that there are better-performing DFR versions, especially in terms of basic models of randomness. Often used candidates include the Bernoulli model of randomness with Poisson approximation and the Bose-Einstein model of randomness. The reason why we have chosen the TF-IDF-based model of randomness is that it closely relates to other atomic retrieval models in the thesis. The focus in this thesis is not on which atomic model performs the best with our proposed field weighting methods, but instead to demonstrate that they can be used with any atomic retrieval models. No significant effort is spent comparing our field weighting method used together with BM25 against it being used with DFR for example. The focus instead is on demonstrating that whatever the underlying atomic model is, the proposed field weighting methods increase performance. For these reasons, it is not important which DFR model is used, which is also why only one is used.

2.4.6 Non-Analytical Supervised Ranking Models

The distinction in literature between different types of ranking models is not a clear one. Probabilistic models are sometimes considered to encompass all the above-mentioned models, even though it is difficult to see the probabilistic foundations of TF-IDF and VSM. Furthermore, once training data and supervised learning enters the equation things get even more muddled. For example, it could be argued that the BM25 and LM are supervised models if their hyperparameters are optimised. As discussed before, in this thesis the distinction between analytical and non-analytical is highly important. Therefore, in terms of notation in this thesis all non-analytical models will be called learning-to-rank (LTR) models, whether they are feature driven, or deep learning-based. This helps clear the confusion between learning (LTR parameters) and hyperparameter tuning (BM25 k_1 and b tuning).

The focus of this thesis is on analytical models as they are a better fit for investigative tasks, which means not as much time will be given to non-analytical models. However, for context, these models are introduced briefly.

Feature Driven Learn-To-Rank Models

An extensive summary of the research on feature-driven learning-to-rank (LTR) models can be found in Liu et al. [41]. We summarise their description of the field and extend it where appropriate.

Where in BM25, LM and DFR a ranking model is defined through conditional probabilities and for TF-IDF through heuristics (to an extent), the idea underlying LTR is to learn the model from the data [41]. There are three main approaches to doing this:

- **Pointwise.** Input = document + query. Output = single value used to rank all documents.

- **Pairwise.** Input = (document1 + query) + (document2 + query). Output: Preference on which document should be ranked higher.

- **Listwise.** Input = list of all documents + query. Output = ranked list of documents.

The first two approaches are often modelled as regression or classification tasks with the corresponding loss functions. The listwise approaches calculate loss more naturally in terms of a ranking task, using accuracy metrics, such as Normalised Discounted Cumulative Gain (NDCG) over the entire ranking directly [42].

Table 2.2 shows classification of feature-driven LTR models from Lie et al. [41] (extended).

	SVM	Boosting	NeuralNet	Others
Point	OC SVM [43]	McRank [44]		Prank [45] Subset Ranking [46]
Pair	Ranking SVM [47] IR SVM [50]	RankBoost [48] GBRank [51] LambdaMart [53]	RankNet [49] Frank [52] LambdaRank [54]	
List	SVM MAP [55] PermuRank [59]	AdaRank [56]	ListNet [57] ListMLE [60]	SoftRank [58] AppRank [61]

Table 2.2: Feature-driven LTR models from and their classification from Liu et al. [41].

All the models in Table 2.2 require manual definition of features. Things like BM25 scores and PageRank scores are often used, but theoretically, any feature defined between a document and a query can be used [41]. Constructing and choosing the features is not a simple task and usually requires knowledge of the data (in terms of structure for example) and of the kinds of queries that users might submit. This means there have to be query logs available. It is also time-consuming as it has to be done by hand [62].

There are two main methods for developing training data for LTR models. 1. Human annotated query-relevant document pairs and 2. click-through data. With regards to the application area of this thesis, i.e. IR for data investigations, each of these approaches has problems. For the first one the issue is that in a more exploratory search scenario, the user's information needs and therefore possible queries are not known (no query logs), which means it is not possible to annotate a representative set of them. For the second one, the problem is even more apparent, as the data investigators work with rarely has been explored by many other people, thus it is unlikely that there is click-through data available. Furthermore, click-through data tends to be proprietary.

Raw-Feature Driven Ranking Models

Deep learning ranking methods do not require constructed features, instead, they use the raw text from queries and documents directly. This differentiates these models from those using neural networks mentioned in the previous section.

[62] provides a good overview of models in this area. We will summarise

some of the different model types, but will not go into further detail as deep learning models are inherently non-analytical and thus do not fit well with the topic of this thesis.

Deep learning approaches in IR tend to be borrowed from other areas of research, such as NLP and classification. However, IR has some unique issues that need to be resolved before these models can be deployed: Firstly, queries and documents are different in length and in nature. Secondly, there exists what is known as the semantic gap between the query and the document, i.e. different terms are used for describing the same thing depending on the context. According to [62] the deep neural components and techniques central to deep IR models are as follows: convolutional neural networks (CNN) [63], recurring neural networks (RNN) [63], Long term-short memory (LSTM) [64], gated recurrent units (GRU) [65], attention mechanisms and word embeddings.

More recently deep learning IR models built on large-scale language models[4] such as BERT, T5 and GPT-3 have become increasingly popular. They have been shown to beat previous approaches by a significant margin [66]. However, using the language models for IR has some drawbacks: Firstly, the models have a maximum length of the term vector (i.e. document length) that can be used, meaning documents have to be short. Secondly, if these models are used directly they are slow. For each new query, an inference step is required, which makes many of the models obsolete in a real-world retrieval scenario. Some approaches chop documents up and aggregate the parts later to deal with the first problem and some avoid the latency issue by loosening the term dependency assumptions in the network [67, 68].

All LTR approaches, whether they are feature-driven, no-feature-driven, or based on large-scale language models tend to trade off accuracy for efficiency. However, in terms of this thesis, there is an even more important trade-off. As these models are not analytical, increases in accuracy cannot be assigned to any formal part of the model, meaning as they become more complex and more accurate, they become less transparent. It is this trade-off between accuracy and transparency that resulted in the focus on analytical models for this thesis.

2.5 Structured Document Retrieval (SDR)

Structured document retrieval (SDR) refers to any retrieval scenario where the objects of interest (documents) are structured and where this structure is used by the retrieval system. In the past, much of the research in this area has focused on hierarchical structures found in document types such as XML. However, the

[4] different from Language Modelling retrieval approach discussed earlier

focus of this thesis is on non-hierarchical structured data. This makes the proposed approaches more general, as hierarchical structures documents can be flattened to non-hierarchical structures, but not vice versa. Furthermore, the retrieval task is “easier” to an extent as the depth dimension is not a problem. Since analytical models for SDR is a research area far from being saturated, it makes sense to start from the “easier” scenario.

2.5.1 Field Notation

To avoid confusion between documents, collections, document fields and collection fields it is worth clarifying the notation used.

- f = document field: e.g. the title of a document.
- F = collection field: e.g. all the titles in the collection.
- $n(t, f)$ = term frequency (in document field f): how many times term t occurs in f.
- $\mathrm{TF}_M(t, f, (F))$ = term frequency quantification: the term frequency component of retrieval model M, with respect to a document field $f/$
- $\mathrm{df}(t, F)$ = document frequency (in collection field F): how many times term t occurs in F.

2.5.2 Document Structures and Information Needs

All the examples I could find of large-scale IDJ projects deal with data sets where structure is important. Whether it is in terms of search, entity linkage, or entity relationships, the data never seems to be atomic. This is likely due to the fact that the underlying data often consists of things like emails, contracts and spreadsheets, all of which are structured. This is one of the main reasons I chose to make document structures a central part of my thesis.

Before diving into existing Structured Document Retrieval (SDR) models, it is worth discussing data structures generally in some detail. Specifically, the different ways in which parts of the structure (document fields) can be connected and how their interplay might help us model information needs.

Consider an ad-hoc retrieval scenario with a single search field over a structured data set with multiple fields. Even though few assumptions are made about the nature of the data in this thesis, it is important to discuss differences in potential structure types for two reasons: 1. the considerations highlight issues that SDR models need to account for and 2. they directly affect the methodology used to model information needs in this thesis.

A common example of a document structure in SDR literature is one with two or three fields that all serve a similar purpose. For example a *title*, *body* and sometimes *anchor text* or *in-link text* [69, 70, 71, 30]. In such cases, the fields can be understood to differ only in their "quality". This idea refers to the concept of elite terms, which describe the topic the author wants to talk about better than other terms [33, 72, 73]. With limited space (titles are short) the author is likely to use more elite terms [69]. Therefore the terms in *title* are better "quality" and their importance should be boosted.

Other SDR scenarios include ones where the fields refer to completely different aspects of the document, such as the *author* or *date of publication.* This category would also include things like product catalogues, with fields such as *manufacturer*, *product name* and *description.* Lastly, more complex structures might include fields such as *titles of related documents.*

In cases where the fields are different representations of the same aspect of the document, such as the {*title*, *body*} example, it is important to account for the dependency of term occurrences across the fields. This is the biggest strength of probably the best-known multi-field retrieval mode; the BM25F [69]. However, there are apparent downsides to assuming strong constant dependence on term occurrences across fields in terms of the other two examples. For example, the dependency between {*title* and *author*} is not as clear as {*title* and *body*}. This is because document titles and bodies are likely to use the same terms to talk about the same entity, meaning the dependence between the term occurrences should be assumed to be high, whereas the author field is likely to have terms which when mentioned in the body, refer to another entity. For example, if John Doe is the name of the author and the term John is mentioned in the body, it is likely that it means another John. Therefore the dependency of terms should be assumed to be lower. Even more problematic would be the dependency between {*title* and *titles of related documents*}. This is because almost by definition the terms in the two fields refer to different entities.

Another aspect of SDR is whether to emphasise the occurrences of terms differently if they appear in many fields, as opposed to one field. In the case of {title and body} it might not matter whether a term appears twice in the title, or once in the title and once in the body, apart from the effect of the field weights. However, in the case of {*title* and *titles of related documents*} it clearly does, as the occurrences by definition refer to different entities.

One of the key aspects of my research is to move away from the notion of treating document structures as a nuisance that needs to be dealt with. A set of widely used probabilistic SDR models handle the nuisance of cross-field term dependency by simply modelling the documents as atomic after some form of field weighting. It will be demonstrated later on that even though these models perform better, due to being less noisy, they miss out on significant potential performance gains, by ignoring structure. The models I introduce consider document structures as useful and something to be leveraged.

2.5.3 SDR Models for Hierarchical Structures

The first in-depth analysis of SDR is by Wilkinson [74]. The work is usually cited to argue that considering the structure of documents, rather than seeing them as atomic, tends to be beneficial for retrieval performance. The paper includes experiments with multiple models, many of which combine document section-based relevance scores using a weighted sum.

With the increased demand for systems that could deal with structured document types such as SGML and XML, came a host of approaches based on different theoretical grounding. Lalmas et al. use Dempster-Shafer Theory of Evidence for modelling uncertainty in SDR instead of probability theory, producing various models that represent document structure as trees [75, 76, 77].They argue that the DS Theory of Evidence is a better fit than probability theory because according to them it is more flexible if dealing with aggregated components [75]. Fuzzy logic has been used by Kazai et al. [78]. Bayesian inference and network methods were used by Myaeng et al. and Piwowarski et al. [79, 80] amongst others. They model documents as hierarchical networks. Other earlier approaches by Baumgarten and Lalmas [81, 82, 83] used probabilistic logic.

Roelleke [84] points out that all the above approaches share the common feature of using a variant of the TF-IDF weighting scheme. Another similarity is that they all consider the document structures as hierarchical. This means that an inherent part of the models is to control how the importance of elements is reflected higher on the tree, i.e. if element E_1 has two sub-elements E_2 and E_3, how is the importance of TF-IDF scores in E_2 and E_3 reflected in E_1 for example. [84] suggests adding an accessibility component to the TF-IDF weighting scheme, where the importance given to terms is discounted when moving up the tree depending on the nature of the document structure. Their model is denoted TF-IDF-acc.

With the of INiative for the Evaluation of XML Retrieval (INEX) in the early 2000s, structured document retrieval started focusing on XML documents specifically. The research looks at querying the XML data both using formal languages such as XPath, XIRQL and XXL as well as using natural language queries [85, 86, 87]. An important variable in deciding which approach to take has to do with how much the user knows about the data structure [88, 89]. The main aim of many XML retrieval approaches is to balance between two types of users, those who understand and know the document structure and can thus express it in their queries and those who do not [89, 90]. Many of the approaches discussed before were used at INEX as well, with the addition of LM-based SDR models in the early and mid 2000s [90, 91].

The task of XML retrieval in terms of INEX is not a simple one to describe compared to ad-hod-retrieval with atomic documents for example. There are a number of variables which need to be considered. Firstly, what part of the document should be shown to the user? In XML retrieval this problem is investigated by determining the best point of entry for a given query [78]. Secondly, as pointed out by Kazai et al. [78] amongst others, depending on the nature of the data (many short elements, or fewer long ones) different approaches should be considered. Some require more focus on the structure of the document, and some on the content. Thirdly, not only do the different elements of the documents need to be considered separately they are also nested, meaning their hierarchical relations need to be taken into account. Fourthly, the degree to which the user is familiar with the document structure has an effect on both the choice of appropriate models and the element types that should be presented. If the user is not familiar with the structure, it is much less likely that their query can be leveraged to recognize correct entry points or constraints on the structure. And finally, a common aspect of many of the XML retrieval models discussed above, especially those more focused on structure, rather than content, is the fact that they consider the semantics of field types. Meaning some expect queries to name the field types, either explicitly through a formal language such as Xpath, or simply in the query (e.g. show me titles of movies with Brad Pitt).

The SDR approach proposed in this thesis considers a more simple scenario than the one described by the above 5 points. Firstly, the question of which parts of documents to show to the user is not a priority. Secondly, documents are modelled as non-hierarchical, meaning each document has m non-nested elements (fields). And finally, no knowledge of the structure by the user is assumed, meaning the queries cannot be expected to contain hints about which elements to focus on. These "simplifications" reflect the characteristics of investigative retrieval. However, it is worth mentioning the third point here. If we cannot use the queries to understand which elements the user is interested in how do we assign weights to them? One approach used extensively by the models in the next section is to optimize weights over the different element types using training data, meaning fields that are known to be important purely from experience are given more weight.

One of the key points of this entire thesis is to argue that this is not what should be done. Karen Spark-Jones argued in the 70s that term specificity weights should be considered a statistical property of terms, not a semantic one set arbitrarily. So too this thesis argues that field weights should not be considered a semantic property of fields learned from training data, but a statistical one set according to collection statistics.

2.5.4 Field Score Aggregation (FSA) v. Term Frequency Aggregation (TFA)

The models in this section can be said to have one of two underlying aggregation functions: one that explicitly considers the structure (FSA), and one that does not (TFA). The following two definitions will explain each approach in detail. These high-level definitions do not consider the specificity of terms (e.g. IDF), only how the score contributions from an increase in term frequencies are considered. Firstly, there are the models that rank documents on each field type, and then aggregate the field-based retrieval scores using weights:

Definition 2.29 (Weighted Sum of Scores (FSA))**.** *Let d be a document consisting of m fields, f be a field consisting of terms and w_f a weight assigned to a field and M be an atomic retrieval model e.g. BM25. In the context of calculating the RSV, fields are considered in the same way as documents were for atomic retrieval.*

$$\mathrm{RSV}_{\mathrm{FSA},M}(d,q,c) := \sum_{i=1}^{m} w_i \cdot \mathrm{RSV}_M(f_i,q,c) \tag{2.49}$$

Equation (2.49) can be interpreted as a special case of the utility function where the utility of a field is replaced by its rank score.

The other group of models — most notably the BM25F — applies the weights to the field-based term frequencies, sums them together across the fields and then retrieves over these aggregated documents.

Definition 2.30 (Weighted Sum of Term Frequencies (TFA))**.** *Let $\vec{f} = [n(t_1,f)\ldots n(t_{|f|},f)]$ be a vector representation of field f, $n(t,f)$ the term frequency of term t in field f and w_f a field weight.*

$$\mathrm{RSV}_{\mathrm{TFA}}(d,q,c) := \mathrm{RSV}(\bar{d},q,c) \tag{2.50}$$

$$\bar{d} := [w_1 \times \vec{f_1} \ldots w_{f_m} \times \vec{f_m}] \tag{2.51}$$

Each of the above aggregation functions has its strengths, although the latter is usually considered more robust. Chapter 4 discusses the strengths and weaknesses at length, whilst demonstrating the cost of these weaknesses using formal retrieval constraints and analysis with real-world data.

2.5.5 Field Score Aggregation (FSA) Models

FSA refers to SDR models where traditional retrieval models are used to score documents based on each field and the scores are aggregated using field weights (See Definition 2.29). The retrieval scores can be produced by any atomic model, BM25 or LM for example. Such models are closely related to the field of meta-search where the scores of different search engines are aggregated to a single ranking [41, 69]. The RSV score for FSA models with different field level models is calculated using Definition 2.29. In terms of notation, an FSA model using BM25 as the underlying atomic retrieval model is denoted FSA-BM25 and its RSV as $\mathrm{RSV}_{\mathrm{LFSA,BM25}}$.

2.5.6 BM25-Field (BM25F)

Robertson el al. introduced the BM25F in order to allow for term frequency saturation across fields. BM25F calculates the retrieval score of a fielded document as the BM25 score of a flattened document representation where the field weights are applied directly to the term frequencies of the fields: $n_{\vec{w}}(t,d) = \sum_i^m w_i n(t, f_i, d)$, meaning the aggregation function is TFA [69, 70]:

Definition 2.31 (BM25F Retrieval Status Value)**.** *Let $n_{\vec{w}}(t,d)$ be the weighted sum of term frequencies over the fields and* rel *is the relevance information used by the RSJ weight.*

$$\mathrm{RSV}_{\mathrm{BM25F}}(d,q,c) := \sum_{t \in q \cap d} \frac{(k_1+1) n_{\vec{w}}(t,d)}{n_{\vec{w}}(t,d) + k_1 \left(b \frac{|d|}{\mathrm{avgdl}(c)} + (1-b) \right)} w_{\mathrm{rsj}}(t,c,\mathrm{rel}) \tag{2.52}$$

In the absence of relevance information the w_{rsj} becomes the IDF [21] as discussed in Section 2.4.2.

Robertson et al.[92] later introduced a version of BM25F where length normalization is applied to each field, rather than the whole document:

$$n_{\vec{w}}(t,d) = \sum_i^m w_i \frac{n(t,f_i,d)}{B(b_i,f_i,F_i)} \tag{2.53}$$

$$B(b_i,f_i,F_i) = \left((1-b_i) + b_i \frac{|f_i|}{\mathrm{avgdl}(F_i)} \right) \tag{2.54}$$

BM25F is considered perhaps the most effective analytical SDR model, both in terms of commercial adaptations (e.g. ElasticSearch, Lucene etc.) and as a baseline in academic research [93, 30, 94]. It is also one of the main ones used as a main benchmark in this thesis.

2.5.7 Other Analytical SDR Approaches

Mixture of Language Models (MLM)

The retrieval score for MLM is calculated by applying the field weights over field-based language models (θ_f), summing the resulting probabilities together and taking their product over the query terms [71].

Definition 2.32 (RSV-MLM). *Let θ_d be the mixed language model, θ_f a field level language model, $\sum_{f=1}^{m} w_{f_i} = 1$ and λ the mixture parameter calculated using dirichlet smoothing.*

$$\mathrm{RSV}_{\mathrm{MLM}}(d, q, c) := \prod_{t \in q} P(t|\theta_d) \tag{2.55}$$

$$P(t|\theta_d) := \sum_{f=1}^{m} w_f P(t|\theta_f) \tag{2.56}$$

$$P(t|\theta_f) := \lambda_1 P(t|f) + \lambda_2 P(t|F) \tag{2.57}$$

The probabilities $P(t|F)$ and $P(t|f)$ are calculated using the maximum likelihood estimate as: $P(t|X) := \frac{n(t,X)}{|X|}$ where $X \in \{f, F\}$.

Since, the field weights are incorporated into the model explicitly, rather than applied over field-based retrieval scores, the underlying aggregation model for the MLM is TFA.

Probabilistic Retrieval Model for Semistructured Data (PRMS)

The PRMS model uses the probability of query terms appearing in fields for better mapping between the two [95].

$$\mathrm{RSV}_{\mathrm{PRMS}}(q, d, c) := P(q|d) := \prod_{t \in q} \sum_{f=1}^{m} P(F_i|t) P(t|f_i) \tag{2.58}$$

where $P(F_i|t)$ is define using Bayes theorem as: $P(F_i|t) = \frac{P(t,f)P(f)}{P(t)}$ and $P(q|f_i) = \frac{\mathrm{TF}(t,f_c)}{|N|}$. See [95] for further details.

The PRMS model is mentioned here as it appears in literature often. However, it is not given much emphasis as both the experimentation in this thesis and in many papers have demonstrated that it does not perform well [94, 96].

Fielded Sequential Dependence Model

The Fielded Sequential Dependence (FSDM) is a multi-field adaptation of the non-fielded Sequential Dependence Model [97, 98, 99]. With MLM as its underlying model, FSDM adds ordered and unordered bigrams to the ranking function, thus modelling for the dependency between words. The ranking score for FSDM is defined as a weighted sum of the ranking scores based on the different term definitions (unigrams, ordered bigrams and unordered bigrams). The weights are usually learned.

2.5.8 Learning-to-Rank and Structured Document Retrieval

With the increased popularity of neural networks (NN) and deep learning in most data-driven research areas, its use in IR has also become more common. Learning-to-Rank (LTR) models have been outperforming traditional models for most retrieval-task types in the last 10 or so years. However, not much emphasis has been given to document structures in the learning-to-rank literature. Non-fielded LTR models are often used as benchmarks against fielded models and vice versa [30, 93]. This thesis focuses on models that are designed specifically to deal with structured data. This is because the end goal of this thesis is not only to increase the accuracy of SDR models but to understand the structure as well. And to do that the model has to consider the structure given the query.

Trabelsi et al. [62] provide an extensive summary of neural models for document retrieval. Their focus is on non-fielded data, but some time is spent on SDR as well. They discuss table retrieval methods at length as they see it as a type of SDR. Table retrieval seeks to match tables to queries [62]. As an area, it is out of the scope of the literature considered here.

Perhaps the best known and definitely the most cited paper on neural models (the only one pointed out by [62]) for SDR at the time of writing this thesis is “Neural Ranking Models with Multiple Document Fields” [93]. This major piece of research by the team at Microsoft and academics at UCL introduces a very complex neural network which is able to consider each field separately and also aggregates them in a non-linear way.

Their model outperforms the benchmarks, including BM25F. However, in terms of NDCG@10, the difference between the two is only about 5%.

Another neural network approach for SDR looks at the domain of Semantic Product Search [100]. The research was conducted on Home Depot data by people working at Microsoft, Home Depot and Emory University. Their model outperforms the baselines including the BM25F in terms of some accuracy metrics but fails to show an improvement in terms of MAP.

Balaneshinkordan et al. [30] introduced the Attention-based Neural Architecture for Ad-hoc Structured Document Retrieval (ANSR) model, which has received less attention but still falls well within the group of neural models for SDR discussed here. They focus on attention gates, however, to compare the query and document representations they use simple cosine similarity, rather than a fully connected network, which was criticised later on by Zamani et al.[93].

The number of LTR methods aimed at structured documents specifically remains relatively low. They are usually aimed at specific tasks such as web search as in the case of Zamanani et al. [93], or product search in the case of Choi et al. [100], with the exception of Balaneshinkordan et al. [30]. Furthermore, the increases in performance are not necessarily as big as one might hope. They tend to do better than the BM25F, which is to be expected as they can tackle things like the semantic gap between the queries and documents. However, it is not clear that they are able to leverage the structure in new ways. The issues discussed in the next chapter are still likely to affect these more complex models.

Another important issue to note is that due to the complexity of neural ranking models, they are usually not trained directly on the entire database. Instead, they are usually trained on candidate documents, i.e. a set of documents retrieved with a model such as the BM25 for non-structured documents, or BM25F for structured documents [62]. This technically makes them re-ranking models, not ranking models and comparing the performance of a re-ranker to the ranker that precedes it, is problematic, to say the least.

2.6 Term Frequency Saturation in IR models

The purpose of this section is to discuss how the retrieval models introduced so far apply term frequency saturation and how this affects their ranking behaviour. This is an important discussion as term frequency saturation and the effects it has on performance, retrieval constraint satisfaction and intuitiveness of rankings are central aspects of this thesis. What term frequency saturation means is that as the number of term occurrences grows, each new one should be given less importance than the last one. For example, if term frequency increases from 100 to 101, this should have a smaller effect on the retrieval score than increasing from 2 to 3. More formally, the second derivative of the TF function should be negative: $\frac{\partial^2 \text{RSV}}{\partial \text{TF}^2} < 0$.

But why is term frequency saturation important? Consider the documents in Table 2.3.

	$n(t_1, d)$	$n(t_2, d)$
d1	1	1
d2	2	0

Table 2.3: Example documents. Equal term specificity assumed, i.e. specificity components e.g. IDF do not affect RSV. Furthermore, assume equal document length. $q = \{t_1, t_2\}$

If term frequency is not saturated, i.e. the second occurrence of t_1 in $d2$ is given the same importance as the first occurrence of t_2 in $d1$ the two documents would receive the same score. This would be the case for any model using TF-IDF with $\mathrm{TF}_{\mathrm{raw}}$(2.9). However, intuitively we would prefer documents where more unique query terms occur. By saturating term frequency and thus giving the second occurrence of t_1 less weight than the first one, and therefore less weight than the occurrence of t_2 in $d1$, we ensure that documents having more unique query terms are ranked higher.

2.6.1 Atomic Document Models

As mentioned in the previous section BM25 has a hyperparameter controlling the degree of term frequency saturation. Figure 2.3 shows how Definitions 2.9, 2.10 and 2.12 affect term frequency saturation across the first 5 term occurrences. Equations (2.14) and (2.16) have been slightly modified so that they both start from 1 in the figure. This modification does not affect their ranking results in any way, but it makes the figure more readable.

$$\log\left(1 + \mathrm{TF}_{\mathrm{raw}}(t, d)\right) \rightarrow \log\left(1 + \mathrm{TF}_{\mathrm{raw}}(t, d)\right) \times \frac{1}{\log(2)} \tag{2.59}$$

$$\frac{\mathrm{TF}_{\mathrm{piv,b}}(t, d, c)}{\mathrm{TF}_{\mathrm{piv,b}}(t, d, c) + k_1} \rightarrow \frac{(k_1 + 1)\,\mathrm{TF}_{\mathrm{piv,b}}(t, d, c)}{\mathrm{TF}_{\mathrm{piv,b}}(t, d, c) + k_1} \tag{2.60}$$

Figure 2.3 shows that by choosing k_1 it is easy to set the degree of term frequency saturation. Furthermore, it can be seen how that degree is quite high for the recommended k_1 range of 1.2 - 2.0.

Consider Table 2.3 but with $n(t_1, d2) = 10$. Should $d1$ still have a higher rank score than $d2$? With $k_1 < 1.2$, this would be the case, but not with $k_1 = 2.0$. For language modelling, there is no hyperparameter to directly adjust the degree of term frequency saturation. However, term frequency is saturated through the log component in Equation (2.21).

Of course, these considerations are only with respect to changes in term frequency, rather than term specificity. If term t_1 had much higher specificity than t_2, depending on k_1 it is possible that document 2 would rank higher, since the second occurrence of t_2 in document 2 would be given more weight than the first occurrence of t_2 in document 1, due to the IDF / RSJ component.

Figure 2.4 demonstrates an important difference between the BM25 and LM models: On the left, we have BM25 models with k_1 set as 1.2 (top) and 2.0 (bottom). On the right, we have LM (Definition 2.16) models with λ set as 0.1 (top) and 0.25 (bottom). The horizontal bar lines show how query terms of different specificities interact. For example, the green line in the top left graph

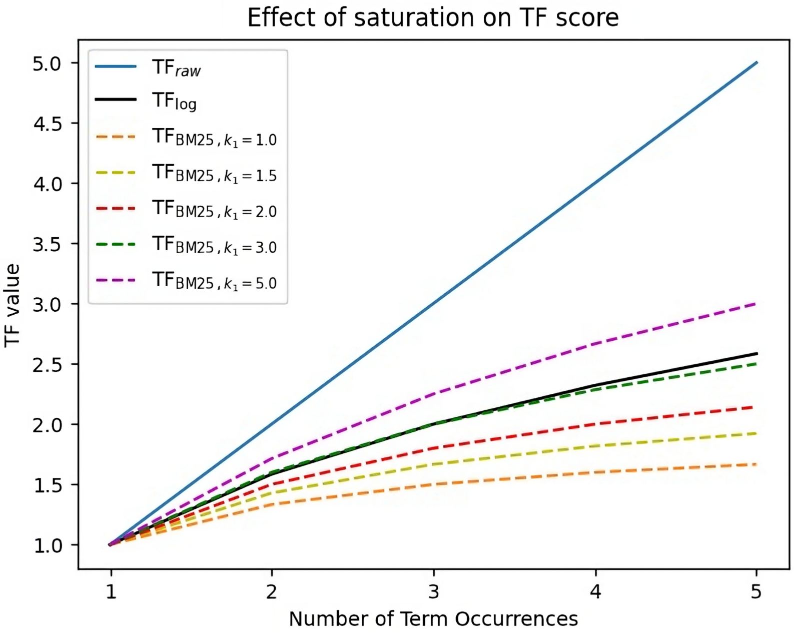


Figure 2.3: Term Frequency Saturation for different term frequency quantifications. Figure inspiration taken from Roelleke [19].

($k_1 = 1.2$) shows the relationship between term frequency and retrieval score for a relatively rare term. The horizontal line meets the green at approximately term frequency = 2.5. This means that for a term with the specificity of the orange line, we would need a term frequency of three in order to out-weight one occurrence of a term with the specificity of the green line. However, if we look at the bottom left graph (k1=2.0) we see that the horizontal bar between the green and orange lines is shorter. Now they meet at term frequency = 2, meaning that two occurrences of the orange term are enough to compensate for one of the green ones. By adjusting the k_1 hyperparameter we can therefore change model behaviour with respect to how terms with different specificities interact. The same is not true for LM. This is evident from the figure as the length of the horizontal bars is the same for all levels of λ and all specificities. The reason for this is that in LM the specificity component, i.e. the background model is within the saturation method, i.e. the log.

With respect to LM [29] points out that “this nonlinearity [saturation of tf] is only the consequence of a mathematical transformation, and the actual dependency between successive occurrences of the same term is not modelled”.

Cummins [29] introduced the Polya Urn Document Language Model (LM-SPUD) in order to model this dependency, i.e. term frequency saturation for-

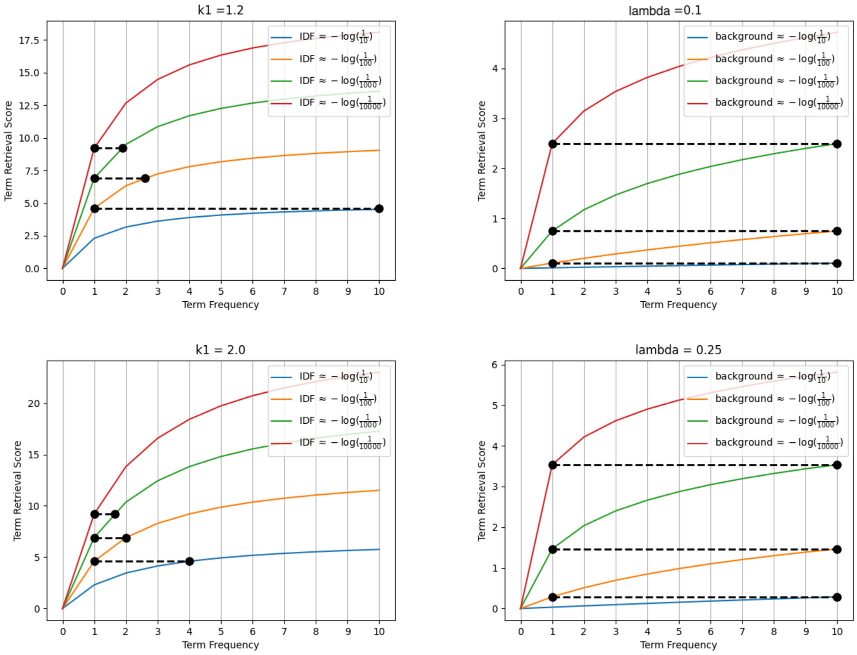


Figure 2.4: The difference in modelling term frequency saturation between BM25 and LM. BM25 on the Left and LM on the right. The main takeaway is that adjusting model parameters has no effect on the relative importance given to terms with different specificities. See below for an in-depth discussion.

mally. However, the model does not contain a hyperparameter for adjusting the degree of term frequency saturation, much like log-TF and LM-Multinomal models. Figure 2.5 shows how term frequency saturation is different between the LM-multinominal and LM-SPUD.

The term frequency saturation in the DFR retrieval model depends on the model of randomness chosen for the base model and the first normalization. However, none of the models of randomness allows for adjusting the degree of term frequency saturation, this remains exclusively possible for the BM25.

Term frequency saturation in atomic models is relevant to this thesis as its demonstrated importance in non-structured retrieval was one of the main inspirations for applying it to structured models in a more formal manner.

2.6.2 Structured Document Models

Term frequency saturation in SDR is more complex than atomic retrieval. The reason for this is made evident by Table 2.4 and the discussion that follows.

Unlike in Table 2.3, here the term occurrences are spread over multiple fields (f_1 and f_2). This brings us back to the question from the beginning of this

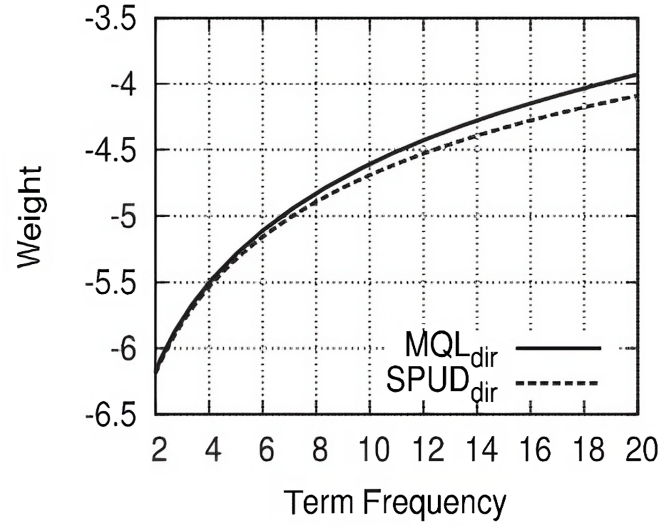


Figure 2.5: The effect of change in TF on term weight for Polya-urn LM model [29]. MQL$_{\text{dir}}$ = multinominal LM with dirichlet-based smoothing, SPUD = Smoothed Polya Urn Document model with dirichlet-based smoothing. Figure from Cummins [29].

field	f_1		f_2		flattened doc
term	$n(t_1, d)$	$n(t_2, d)$	$n(t_1, d)$	$n(t_2, d)$	
d1	1	0	0	1	$t_1 + t_2$
d2	1	0	1	0	$t_1 + t_1$

Table 2.4: Example with two fields f_1, f_2 and two query terms t_1, t_2 illustrating term frequency saturation across fields.

section about how should term occurrence dependencies, i.e. their saturation be modelled across fields. The two occurrences of a term are likely to be highly dependent if the fields are *title* and *body* for example. The notion of the fields only differing in quality by Robertson et al. [69] would be valid here. But what if the fields are titles and related titles, for example, it is less clear whether dependency can be assumed.

FSA-based SDR models do not model the dependency across fields at all. Since the field-based scores (RSV(q, f)) are calculated independently without considering other fields, the term frequency of term t in field f_1 is assumed to be independent of its term frequency in any other field. In a sense, FSA models end up double counting the second occurrence of term t_1 in Table 2.4. This double counting means that given equal specificity weights, the two documents in Table 2.4 have the same ranking score, even though one of them has more unique query terms.

Robertson et al. was the first to formally discuss this issue at length and to provide a solution with their BM25F approach, albeit the MLM model was introduced sooner and also does not suffer from it [69, 71]. Since both of these models concatenate the fields into flattened document presentation before calculating the final score (TFA-based models), term frequency is saturated in exactly the same way as it is for the atomic counterparts. For the BM25F this is evident from Definition 2.31, as the BM25-TF is calculated over flattened document representation, meaning term frequency saturation is applied. This in turn means that the documents in Table 2.4 are ranked intuitively. TFA-based models thus saturate term frequency across fields, but in doing so revert back to considering the documents as non-structured. Chapter 4 discusses and demonstrates why this is problematic and Chapter 5 proposes more advanced methods for cross-field term frequency saturation.

2.7 Axiomatic Retrieval and Retrieval Constraints

Fang et al. [26] introduced formal constraints (axioms) for (atomic) IR, to "capture retrieval heuristics, such as the TF-IDF, in a formal way, making it possible to apply them to any retrieval formula analytically". They extended this work to axiomatic retrieval, where the constraints are used to develop new retrieval models [27]. They also developed semantic retrieval constraints [101].

Their work represents an important branch of IR research for the following reasons: 1. it facilitates model evaluation not only based on accuracy metrics but their underlying behaviour as well, 2. it offers a new starting point for novel model development and 3. the constraints can be used to optimise LTR models more efficiently [102, 32].

Between the original paper ([26]) and the summary paper for retrieval constraints ([28]) the notation and formalization of the constraints changed. For clarity, we summarise the relevant constraints here using the notation from the "summary paper" (Diagnostic Evaluation of Information Retrieval Models) [28].

Table 2.5 shows the formal constraints by Fang et al. [28] and their underlying intuition. As discussed earlier the focus in this thesis is on term frequency and specificity, rather than document length. Therefore only the first four constraints are of interest and will be described here in detail.

Definition 2.33 (TFC1). *Let $q = t$ be a query with only one term t. Assume $|d_1| = |d_2|$.*

$$\forall q, d_1, d_2 \textit{ if } n(t, d_1) > n(t, d_2) \textit{ then } \mathrm{RSV}(q, d_1) > \mathrm{RSV}(q, d_2) \qquad (2.61)$$

Constraints	Intuition
TFC1	to favour a document with more occurrences of a query term
TFC2	to ensure that the amount of increase in score due to adding a query term repeatedly must decrease as more terms are added
TFC3	to favour a document matching more distinct query terms
TDC	to penalize the words popular in the collection and assign higher weights to discriminative terms
LNC1	to penalize a long document (assuming equal TF)
LNC2, TF-LNC	to avoid over-penalizing a long document
TF-LNC	to regulate the interaction of TF and document length

Table 2.5: Heuristic retrieval constraints and their intuition by Fang et. al [28].

The first of the constraints is the most straightforward one, it simply states that all else being equal if term frequency increases in a document, so should the retrieval score for that document. More formally the first derivative of RSV with regards to term frequency should be positive: $\frac{\partial\,\mathrm{RSV}}{\partial\,\mathrm{TF}} > 0$.

Definition 2.34 (TFC2)**.** *Let $q = t$ be a query with only one term t. Assume $|d_1| = |d_2| = |d_3|$ and $n(t, d_1) > 0$.*

$$\begin{aligned} &\forall q, d_1, d_2 \text{ if } n(t, d_2) - n(t, d_1) = 1 \text{ and } n(t, d_3) - n(t, d_2) = 1 \\ &\text{then } \mathrm{RSV}(q, d_2) - \mathrm{RSV}(q, d_1) > \mathrm{RSV}(q, d_3) - \mathrm{RSV}(q, d_2) \end{aligned} \tag{2.62}$$

TFC2 relates perhaps the most to the topics in this thesis. It states that term frequency should be saturated, or more formally the second derivative of RSV with regards to term frequency should be negative: $\frac{\partial^2\,\mathrm{RSV}}{\partial\,\mathrm{TF}^2} < 0$.

Definition 2.35 (TFC3)**.** *Let $q = \{t_1, t_2\}$ be a query consisting of two terms t_1 and t_2. Assume $|d_1| = |d_2| = |d_3|$ and $\mathrm{td}(t_1) = \mathrm{td}(t_2)$ where td is any reasonable measure of term discrimination, i.e. specificity (e.g. IDF).*

$$\forall q, d_1, d_2 \text{ if } n(t_1, d_1) = n(t_1, d_2) + n(t_2, d_2) \text{ and} \tag{2.63}$$

$$n(t_2, d_1) = 0, n(t_1, d_2) \neq 0, n(t_2, d_2) \neq 0 \tag{2.64}$$

$$\text{then } \mathrm{RSV}(q, d_1) < \mathrm{RSV}(q, d_2) \tag{2.65}$$

TFC3 states that given equal specificities for two terms, i.e. IDFs for example, documents with more distinct query terms should be favoured. As discussed in the previous section, TFC3 is closely related to TFC2, as the degree of term frequency saturation, i.e. the size of the second derivative of RSV with regards to term frequency, determines what kind of specificity values TFC3 is satisfied for.

Definition 2.36 (TDC)**.** *Let $q = \{t_1, t_2\}$ be a query consisting of two terms t_1 and t_2. Assume $|d_1| = |d_2|$ and that d_1 only contains t_1 and d_2 only contains t_2.*

$$\forall q, d_1, d_2 \textit{ if } \mathrm{td}(t_1) > \mathrm{td}(t_2) \textit{ then } \mathrm{RSV}(q, d_1) > \mathrm{RSV}(q, d_2) \tag{2.66}$$

The TDC constraint ensures that occurrences of terms with higher term discrimination (e.g. higher IDF) are given more weight.

Axiomatic retrieval and the above retrieval constraints represent an important part of IR research, that directly affects the contributions of this thesis. Chapter 4 introduces constraints for SDR, which in many ways mirror the work done by Fang et al. As has been the case for atomic retrieval, these constraints can be used to develop analytical models (as done in Chapter 5) and to optimize LTR models for SDR.

2.8 Key Components, Similarities and Differences of Atomic and Structured Models

Table 2.6 summarises how atomic and structured retrieval relates to some of the key concepts in this thesis: exhaustivity, specificity, term frequency saturation and retrieval constraints.

One of the key objectives of this thesis is to transfer the lessons learned in analytical atomic retrieval to analytical structured retrieval. To a large extent, this is done by borrowing concepts from the former and applying them to the latter. For example, exhaustivity is a concept as old as IR itself, however, in SDR it is not clear how it should be defined. Should it be the field-based retrieval score, the field-based term frequencies, or something else entirely? The same goes for specificity; ever since Spark-Jones argued that it should be a statistical property — rather than a semantic one — this has not been contested much. In SDR it is difficult to define specificity. One possible option pursued in this thesis is to define it as the weight given to different document fields. Term frequency saturation has been identified as a key issue in both atomic retrieval and structured retrieval. However, currently, it is deployed in SDR by — to an extent — dismissing structure altogether, which in the author's opinion is counterintuitive, as it does not make sense to throw away useful information. This is something this thesis hopes to rectify. retrieval constraints have played a key part in atomic retrieval. Research presented in this thesis introduces such constraints for SDR.

Aspect	Atomic Retrieval	Structured Retrieval
Exhaustivity	The degree to which a term describes a document. Usually estimated from term frequency an appropriate function such as TF-BM25, Log-TF, or foreground model.	**TFA models:** exhaustivity is estimated from summed term frequencies in the same way as atomic models. **FSA models:** exhaustivity can be seen as the field-specific retrieval score.
Specificity	How specific a term is in the collection. Usually estimated from the rareness of the term. Common methods include the IDF and $P(t\|c)$	The part specificity plays in SDR will be discussed at length in Chapter 3. A possible interpretation is that specificity is defined by the field weights.
Term Frequency Saturation	Key component in established models. Handled by taking the log of term frequency in some way (LM, Log-TF), or by other normalization methods (BM25, DFR)	Lack of term frequency saturation in FSA models was one of the underlying motivations for the BM25F [69]. One of the key contributions of this thesis is to incorporate it into the FSA.
retrieval constraints	Formal constraints introduced by Fang et al. [26] allow for the analytical evaluation of models, thus extending the understanding of how models function.	Chapter 4 introduces four constraints for SDR.

Table 2.6: Summarises key aspects of retrieval models. Comparison between atomic and structured models.

Chapter 3

Information Content-based Field Weighting (ICFW)

The purpose of this chapter is to introduce one of the core contributions of this thesis; the use of information content-based field weighting (ICFW) in SDR. Some parts of its contents were published in the BIRD@SIGIR workshop in 2020 [1]. The chapter is structured as follows:

- Section 3.1 describes the motivation of the chapter.
- Section 3.2 introduces the contributions of the chapter.
- Section 3.3 provides intuitive and theoretical justifications for the proposed approach.
- Section 3.4 discusses the context of the proposed method with regard to existing methods.
- Section 3.5 studies of the proposed method with a formal evaluation using benchmark data collections.
- Section 3.6 concludes the chapter.

3.1 Motivation

As discussed in the Background chapter (Chapter 2), the data collections InvIR deals with can be incredibly complex. Much of the pre-processing effort of the data has to do with parsing the structure of the data into a well-defined ontology. In the context of this thesis, the important takeaway is that the data collections used in these investigations tend to have complex structures. Most established

SDR methods tend to leverage document structures by optimising field weights over the document structure, e.g. by boosting the title of the document as it is perceived to be more important than the body field for example. Unless done purely based on prior expert knowledge, these methods require training data in order to optimise the field weights. In investigative scenarios, the data collection would rarely have this kind of training data. If the data collection comes from a leak or is public data we would expect not to have click-through data. With this in mind, this chapter introduces a method for boosting important fields automatically without the need for training data and optimization.

3.2 Introduction

The inspiration for how to automatically set field weights comes from atomic retrieval and more specifically the IDF. The IDF emphasises terms that carry more information and are thus better discriminators between documents. In the same way, the proposed approach uses the amount of information carried by a document field to boost its effect on the final retrieval score. The justification for the proposed method is closely related to the IDF, both in terms of intuition and theory.

In the context of existing SDR approaches the proposed method sits between earlier approaches that considered document structure explicitly (e.g. INEX related models) and more recent adaptations of atomic approaches to SDR (e.g. BM25F). This chapter introduces the BM25-FIC; the first iteration of information content-based field weighting (ICFW), which is the main model of this thesis. BM25-FIC does not saturate term frequency across fields, nor is there any analytical evaluation of the model at this stage. Iteratively developing the ICFW method is one of the main contributions of this thesis.

3.3 Intuitive and Theoretical Justification

The intuitive justification for the use of information content for fields weighting goes back to the work of Spark-Jones from 1972 and leans on the underlying idea of one of the best-known IR concepts; the IDF [20].

There are two proposed theoretical justifications for the use of information content in field weighting. Firstly, information theoretical definitions of the IDF by Aizawa et al. [103] are borrowed to explain the use of the negative log as a measure of information content. Secondly, semantic information theory by [40] is used to justify measures of information content in a similar way as it has been used in the DFR retrieval model.

3.3.1 Intuitive Justification

The inspiration and intuition for the use of information content for field weighting is related to the concept of specificity in atomic retrieval, as discussed in Section 2.8. That is, the underlying conceptual model for how documents fields are weighted is related to a conceptual model from atomic retrieval based on the exhaustivity and specificity of terms: It can be argued that the analytical models described in Section 2.4 (Non-Strucured Document Retrieval) at their core combine the exhaustivity and specificity of a term to give it a weight, that is then summed over the terms in the query. These two concepts are defined here formally for clarity.

Definition 3.1 (Exhaustivity of a Term in a Document)**.** *The degree to which a term describes a document in terms of a query. Usually estimated using some form of term frequency quantification.*

Definition 3.2 (Specificity of a Term in a Collection)**.** *How specific a term is in the collection. Usually measured in terms of how the rarity of a term.*

For TF-IDF and BM25 exhaustivity is represented by the TF quantification and specificity by the IDF / RSJ-weight. For LM the same is true for the foreground model and the background model. For DFR exhaustivity and specificity are more mixed within the Inf1 and Inf2 components. It is our intention to extend this line of thinking to SDR: At the core of most well-known analytical SDR models (BM25, MLM, FSA) are field weights that are adjusted according to the importance of a field. The conceptual model for SDR introduced in this thesis treats a retrieval score for a specific field as a measure of its exhaustivity and the field weight as a measure of its specificity.

Definition 3.3 (Exhaustivity of a Document Field in a Document)**.** *The degree to which a document field describes the document's relevance in terms of the query. Here estimated as the field-specific retrieval score.*

Definition 3.4 (Specificity of a Document Field in a Collection)**.** *How specific a document field is in terms of the collection field. How to best estimate this is one of the core research questions in this thesis.*

Equations (3.1) and (3.2) demonstrate clearly the similarity of the conceptual models between atomic retrieval model BM25[1] and our proposed SDR approach.

Similarity Between BM25 and Proposed Approach

Let d be a document, q a query, c a collection, TF the term frequency quantification, IDF the inverse document frequency, w_f be the field weight of field f,

[1] BM25 used instead of the traditional TF-IDF, as the former is a simple document scoring model, rather than a VSM model.

M an atomic retrieval model e.g. BM25 and $\mathrm{RSV}_{\mathrm{pro}}$ the RSV score for the proposed SDR approach.

$$\mathrm{RSV}_{\mathrm{BM25}}(d,q,c) := \sum_{t \in q} \mathrm{TF}_{\mathrm{BM25}}(t,d) \cdot \mathrm{IDF}(t,c) \sim \sum_{t \in q} \mathrm{exh}(t,d) \cdot \mathrm{spe}(t,c) \tag{3.1}$$

$$\mathrm{RSV}_{\mathrm{pro}}(d,q,c) := \sum_{f \in d} \mathrm{RSV}_{M}(f,q,F) \cdot w_f(f,q,F) \sim \sum_{f \in d} \mathrm{exh}(f,d) \cdot \mathrm{spe}(f,F,q) \tag{3.2}$$

Both Equations 3.1 and Equation 3.2 combine the exhaustivity and specificity, of terms in documents and fields in documents respectively. Where the atomic approach (above) considers the term frequency to reflect the exhaustivity of a term in a document the proposed approach considers the field-based retrieval score to do the same. Specificity is reflected by the IDF in the atomic case and the field weight in the structured case.

Before the development of the IDF, the specificity of a term was often defined semantically [20]. For example "beer" and "tea" would have a higher specificity than "beverage" as they are more specific descriptions of a drink than "beverage". In their seminal work from 1972 Spark-Jones argues that:

> It is not enough, in other words, to think of index term specificity solely in setting up an index vocabulary, as having to do with accuracy of concept representation. We should think of specificity as a function of term use. It should be interpreted as a statistical rather than semantic property of index terms. [20]

Meaning the specificity of a term should be calculated from collection statistics rather than defined semantically [20]. In widely used SDR models (FSA, BM25F, MLM, FSDM) the weight given to a field, i.e. the specificity of a field is usually[2] set semantically, e.g. a title is given more weight because it is known to carry more information, either based on expert knowledge, or training data.

Inspired by the arguments of Spark-Jones for term specificity from 50 years ago, this thesis argues that the specificity of a document field, i.e. field weight "should be interpreted as a statistical, rather than semantic property" of document fields [20]. From this follows that field weights would be interpreted as discriminating features rather than just semantic importance boosting features, i.e. they are used to give more emphasis to document fields which carry more information in a statistical sense, rather than a semantic one.

It is worth noting that in Equation 3.2 the specificity function takes f as an input. This means the specificity is not set for the entire field as a whole,

[2] PRMS is an exception

meaning the specificity of document fields can vary between documents, e.g. the specificity of a title can be different for two different documents. Furthermore, q is also present in both the exhaustivity and specificity terms, meaning these concepts are defined based on the query.

3.3.2 Theoretical Justification

Having established the intuition of what we wish to do with information content-based field weighting, the section will now turn to its theoretical grounding. It is not the intention of this section to formally and unequivocally explain the use of information content for field weighting in terms of mathematical concepts such as probability theory. The following justifications still has gaps and issues. However, the same is true for many popular IR concepts and approaches, including perhaps the best-known IR concept outside of the field; the IDF. Therefore the purpose of this section is to justify the use of information content for field weighting by proposing possible theoretical explanations for it, which hopefully — together with the intuitive justification above — will convince the reader of the validity of the approach.

As the intuition underlying the proposed field weighting method is closely related to TF-IDF and BM25, it makes sense to start its theoretical grounding there as well. As mentioned above, the exhaustivity of a document field with respect to a query is modelled by its retrieval score, making this part of the model straightforward. How about the specificity of a document field? In atomic retrieval — more specifically the TF-IDF and BM25 — specificity corresponds to the IDF / RSJ-weight of a term. To establish a theoretical grounding for statistically calculating a value for the specificity of a document field with respect to a query, we therefore start with the IDF.

The task of establishing theoretical foundations for the IDF is an open question. Many approaches have been suggested, but none are unanimously agreed upon. It is not in the scope of this thesis to summarise all these approaches and the critique they have received. Instead, we focus on the ones that are relevant to this thesis, namely the approach described in Section 2.4.2 by Robertson et al. where the IDF is derived from the BIR model and the approach by Aizawa et al. [103] where the IDF is explained in terms of information theory and mutual information. The former was already described in Section 2.4.2 and related more to the definition of IDF used in this thesis. The latter is described here, as it relates directly to the theoretical grounding for using information content in field weighting.

Aizawa et al.[103] seek to define the TF-IDF model and therefore the IDF as the mutual information between the events of documents occurring and terms occurring within a document collection, ending up with the definition below:

Definition 3.5 (Aizawa's Expected Mutual Information)**.** *Let $E(\mathrm{MI})$ be mutual information and N the total number of documents.*

$$E[\mathrm{MI}(d,c)] := \sum_{t\in d} \frac{n(t,c)}{\sum_{t\in c} n(t,c)} \log \frac{N}{\mathrm{df}(t,c)} \tag{3.3}$$

From Definition 3.5 [103] infers the IDF using a set of assumptions contested by Robertson [21] who points out that the assumptions effectively put the TF-IDF back in the realm of heuristics. The key issue in terms of this thesis is not whether the derivation of TF-IDF by Aizawa et al. [103] is formally correct. Whether or not this is the case, it represents one possible explanation for the IDF, an explanation which this thesis can use to justify using the negative log probability when defining information content in Definition 3.4: $\mathrm{spe(f, F) = \text{-}\,log(P(f, Fq))}$. As the IDF, defined as the negative log of a document containing term t in the above definition, represents the specificity of a term in a collection, we define the specificity of a document field as the negative log of the probability of it occurring, given a query. This line of thought opens up the theoretical justification of our approach to a host of critiques, much like the justification of TF-IDF by Aizawa [103].

To avoid this criticism it is worth pursuing another avenue for justifying using the negative log probability for estimating the specificity of a document field. Instead of defining information content using information theory, it can be defined in a more flexible manner. Hintikka [40] separated approaches for discussing information mathematically to *statistical information theory*, which is information theory as it is usually discussed today with concepts such as entropy, mutual information and information content and to what is called *semantic information theory*, where the information carried by an event can be described in a more flexible manner. According to Hintikka [40], the main difference between statistical information theory and semantic information theory is their approach to probability and uncertainty. Whereas statistical information theory deals with uncertainty in the sense of what happens in the long run in situations that can be repeated again and again, semantic information theory is more concerned with uncertainty in a "logical" sense :

> In a theory of semantic information, we are primarily interested in the different alternatives which one can distinguish from each other by means of the resources of expression we have at our disposal. The more of these alternatives a sentence admits of, the more probable it is in some 'purely logical' sense of the word. [40].

One of the ways semantic information theory portrays information content (IC) of event x is $\mathrm{IC}(x) = 1 - P(x)$, which is famously used by the DFR model to describe information content in their Inf2 component. Here information content is seen as a measure of the risk of accepting Inf1 as a good estimator of the field weight, where risk is explained in the context of utility theory, as in high risk equals high potential gain. Another way Hintikka [40] describes information content, or informative content (the terms are used interchangeably) is $\mathrm{IC}(x) := -\log(P(x))$ (also used by Amati et al. [39]). The justification for the use of these definitions is not grounded in probability theory in the same way as what Hintikka [40] would call statistical information theory, but they do have a logical and intuitive root. Furthermore, they are used to justify the DFR model, meaning there is precedent for their use in IR.

Justifying the combination of exhaustivity and specificity, i.e. retrieval score and field specificity is not attempted in this thesis. However, as we have seen with the TF-IDF, it is not always necessary for a model to be fully grounded in mathematics for it to be used in IR. In fact, the connection between the two components in our approaches is as well defined as the connection between TF and IDF in the TF-IDF approach (Or BM25-TF and IDF). The TF-IDF and BM25 were the inspiration for combining the exhaustivity (field-based retrieval score) and specificity (information content weight) in our approach as well. Other possibilities include using mixture models, as done by LM, although this is not covered in this thesis.

To summarise, the information content of a document field f in this thesis is defined as the negative log of the probability of said document field given a query q and collection c.

Definition 3.6 (Information Content)**.**

$$\mathrm{IC}(x) := -\log(P(f|q,c)) \tag{3.4}$$

The use of negative log can be justified to an extent either through information theory in a similar way as done by Aizawa et al. [103] for the IDF or using semantic information theory as done by [39] in the case of DFR. The analysis and explanation here will be extended in Chapter 5 where the need to combine multiple information content sources arises.

3.4 Background and Context

Having discussed the intuitive and theoretical justification for using information content for field weighting, the chapter will now turn to placing the proposed approach in the context of existing SDR models. These existing models are roughly divided into two parts, earlier models that are explicitly designed for structured (usually hierarchical) data and models that are extensions of atomic retrieval models. The same categorization was used in Chapter 2 (Background). How the proposed approach relates to non-analytical SDR approaches is not discussed here for the reasons described in Chapter 2.

3.4.1 SDR Models for Hierarchical Structures (INEX, XML, SGML)

To recap, Chapter 2 (Background) introduced the INEX, XML and SGML-related models, discussing how they often deal with highly complex data structures (upto 190+ unique document fields) and how the focus on file types such as XML, meant that the models were quite structure specific. In doing so, five distinguishing aspects of this group of models were pointed out. These five aspects are important when considering how our proposed approach is similar to and different to theirs.

- **Point of Entry:** Many of the models give significant emphasis to what part of the document should be displayed to the user in the ranking.
- **Complexity of Data:** The models often make assumptions regarding the nature of the data, i.e. many short and nested elements v. few longer ones.
- **Hierarchy:** Most of the approaches model the documents as trees with more than two levels.
- **Users familiarity with the data:** The degree to which the user knows the structure affects the way in which they query the data.
- **Field type semantics:** The names of the field can give hints about the content. This information is often used by the models.

The approaches in this thesis assume less complex data structures (descibed below) and a more straightforward retrieval scenario than what is described above. There are two main reasons for this, the first one has to do with the data structure and the second one with the retrieval scenario.

Firstly, as the underlying motivation for the thesis was to develop approaches that work with a large variety of data structures in order to be useful in investigations, assuming fewer fields and effectively no hierarchy in sensible. Hierarchical data can always be represented in a flattened way, but not the other way around. The structure assumed here is one where a document has m fields, each a direct sub-component of the document with no hierarchy, or relation between

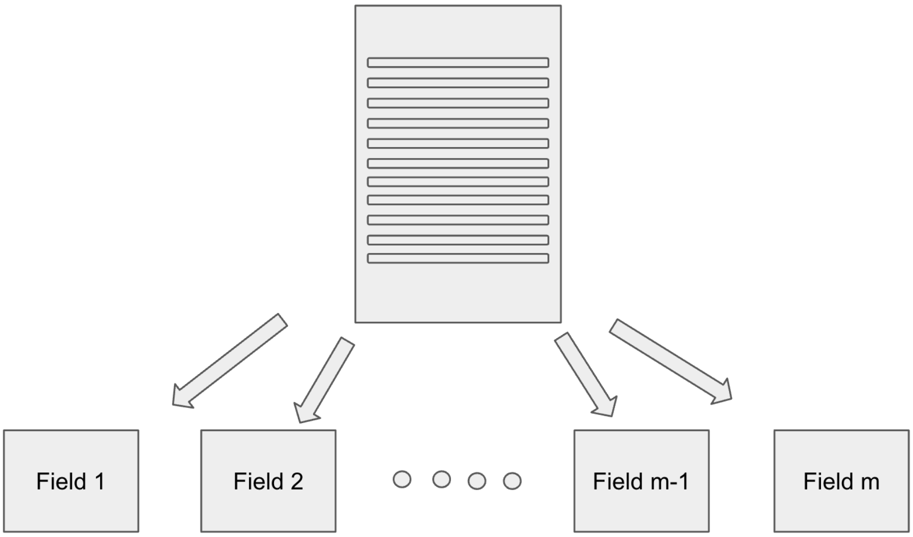


Figure 3.1: Document model

the fields. See Figure 3.1. There is no reason why the approaches developed here could not be extended to hierarchical data. However, this is left for future research. Secondly, the approaches developed in this thesis assume that the user has little, or no knowledge of the data structure to start with. This assumption is made because of the intended application area of the methods, i.e. investigative IR and more specifically IDJ. When digging through new datasets, investigators would rarely be very familiar with their structure, therefore it makes sense to extend this assumption to the retrieval models themselves. What this means is that the semantics of the fields are not used for retrieval purposes, i.e. title is just a field rather than "a short description of the whole document" for example.

A major characteristic of the methods proposed in this thesis and those described above is the emphasis on considering term occurrences in different fields separately. Hierarchy-based models tend to concatenate document elements upwards, thus appreciating the occurrences of terms in separate fields. As we will see in the next section, this is not done by fielded versions of atomic models. A good example of an attempt to appreciate term occurrences in different fields is provided by Roelleke [84], where the occurrences are weighted depending on the nature of the document structure. This line of research was extended by Wang et al. [104] by using context-specific term metrics to calculate the importance of terms up document trees.

To summarise, the main difference between the INEX-related SDR models and the approach proposed in this thesis is their assumptions of the structure of the data and the complexity of the retrieval scenario. The main similarity is their appreciation of term occurrences in multiple fields.

3.4.2 Fielded Atomic Models

As discussed in Chapter 2 (Motivation and Background), after the SDR models described in the previous section, the interest of the research community turned to fielded versions of atomic retrieval models. First with the Mixture of Language Models (MLM) and followed by the various versions of BM25 [71, 69, 70]. The FSDM model was also discussed briefly [99]. However, not too much time is spent on it as it differs from the other models considerably in how it models term occurrences in general. Whereas, the proposed models in this thesis assume that the order in which words occur does not matter e.i. Bag-of-Words (BOW), FSDM uses bigrams and trigrams to model term dependencies.

One of the main differences between MLM and BM25F is that they do not assume term occurrences to be independent across fields. They accomplish this by applying the field weights to the term frequencies directly, rather than to field-based scores as done by FSA models and most of the approaches in INEX and their predecessors. The field weights in these models are set by heuristics (e.g. title gets more weight because it is known to be important), or they are learned from training data. Without these field weights the models effectively revert back to atomic models, as the fields are not considered separately at all (apart from document length normalization for the later version of BM25F [70]).

To summarize, the main difference between this group of SDR models and our proposed approach is that, even without heuristics and training data, our models are able to consider and leverage the document structure, unlike the MLM and BM25F models. What this thesis takes from the fielded atomic models is the emphasis on the importance of cross-field term frequency saturation, which is the focus of Chapter 5.

3.4.3 Other Models

There are two retrieval models worth discussing that do not fall within the categorization of the sections above. These are the PRMS model and its successor the Field Relevance Model (FRM) [95, 105]. The underlying motivation behind PRMS is to boost the term-level probabilities based on the probability of that term occurring in a field. To recap the definition for $\mathrm{RSV}_{\mathrm{PRMS}}$ from Chapter 2.

Definition 3.7. *Let i be a given field and $P_{\mathrm{Ma}}(F_i|t_i) = \frac{P(t_i|F_j,C))}{\sum_{F_k \in F} P(t_i|F_k,C))}$*

$$\mathrm{RSV}_{\mathrm{PRMS}}(d,q,c) := \prod_{i=1}^{n} \sum_{j=1}^{m} P_{\mathrm{Ma}}(F_i|t_i) P(t_i|F_j,d) \tag{3.5}$$

This is related to our proposed SDR approaches in the sense that the probability of terms in a field is used to boost the term-based scores. However, even though the definitions are similar, the underlying motivation is very different: whereas the PRMS boosts terms in fields they are more likely to occur in, information content-based field weighting does the exact opposite, awarding terms in fields where they are rare. The experimentation later in the thesis will demonstrate clearly that ours is the better approach.

The FRM in an extension of the PRMS model which uses known relevance data to boost terms in fields even more if they are known to be relevant. Therefore it is closely related to the RSJ-weight if it is used to incorporate relevance data [92].

3.4.4 Discussion

To summarise the message of the last two sections, the SDR approaches introduced in this thesis are inspired by the conceptual model for atomic retrieval where the weight given to a term is defined by its exhaustivity and specificity. In our approach, the weight given to a document field is defined as a product of its exhaustivity (field-specific retrieval score) and specificity (field weight). As was argued by Spark-Jones 50 years ago with regard to atomic retrieval, the underlying argument for using information content for weighting fields is that specificity should be a statistical property of a document field, rather than a semantic one [20].

A formal justification for the conceptual model and the use of information content for field weighting is attempted using two different approaches. Firstly, information theory is used in a similar way as it has been used for justification of the IDF by Aizawa [103]. Secondly, we discuss a different definition of information content as it was put forward by Hintikka [40]. It is not the intention of this section to unequivocally justify our approach in a formal mathematical way, but rather give intuitive explanations for why it does work.

In terms of existing SDR models, our approach sits between earlier methods where the document structures and their semantics were used explicitly (INEX, XML retrieval, SGML retrieval, database retrieval) and newer methods that are fielded extensions of atomic models. The proposed approach borrows the explicit leveraging of structures (not the semantic aspect) from the earlier models and the use of term frequency saturation from the latter ones, although the latter aspect of the model is not visited until Chapter 5.

3.5 BM25-FIC

The purpose of this study is to demonstrate the usefulness of the use of information content for weighting in SDR and the idea of using statistical field weights, rather than semantic ones in general. The model itself is quite “simple” and therefore we expect it not to always work. However, the lessons learned from this study directly motivated the content of the next Chapter.

3.5.1 Model Specification

To recap, the RSV score for our proposed model at the highest level was defined as Equation (3.2):

$$\begin{aligned}&\mathrm{RSV}_{\text{proposed}}(d,q,c) := \\ &\qquad \sum_{f \in d} \mathrm{RSV}_M(f,q,F) \cdot w_f(f,q,F) \sim \sum_{f \in d} \mathrm{exh}(f,q,F) \cdot \mathrm{spe}(f,q,F)\end{aligned} \tag{3.6}$$

To make the model explicitly about the BM25 we set $M = \mathrm{BM25}$ and rewrite the equation above in terms of the number of fields m.

Definition 3.8 (RSV B25-Field-Information Content (BM25-FIC))**.**

$$\mathrm{RSV}_{\text{BM25-FIC}}(d,q,c) := \sum_{i=1}^{m} w_{f_i}(f_i,q,F_i)\, \mathrm{RSV}_{\mathrm{BM25}}(f,q,F) \tag{3.7}$$

Now the only unknown parameter in the ranking function is the field weight w_f. As discussed previously in this chapter, this weight is defined as the amount of information a document field carries with respect to a query, i.e. its information content. The information content is defined as the negative log of the probability of query q and document field f given a collection field F. Intuitively this means that more weight is given to document fields that are good discriminators.

Definition 3.9 (Field Weight as Information Content)**.**

$$w_{fi}(f_i,q,c) = \mathrm{IC}(f_i,q,c) := -\log(P(q,f_i|F_i)) \tag{3.8}$$

Definition 3.10 (Probability of a Query and Document Field)**.**

$$P(q,f_i|F_i) = \prod_{t \in q \cap f_i} P(t \in f_i|F_i) \tag{3.9}$$

As was the case for DFR, instead of using the probability of a term occurring in a document ($P(t \in f_i|F_i)$), we could also use the probability of a term occurring $n(t,d)$ times in a document ($P(n(t,d|F_i))$), or even the probability of a term occurring $n(t,d)$ times or less in a document (i.e. the CDF). However, focusing on term metrics that are defined for the whole corpus rather than specific documents ($\mathrm{df}(t,c)$ vs. $n(t,d)$) is beneficial from an implementation point of view, as well as a theoretical point of view later on in the thesis when we combine different information features together.

Definition 3.11 (Probability of a Term)**.** *Let $\mathscr{N}$ be the number of documents in which term t could potentially occur, i.e. those that are not empty.*

$$P(t \in f_i|F_i) := \frac{\mathrm{df}(t,f,c)}{\mathscr{N}(t,f,c)} \tag{3.10}$$

According to the log rules and using Equations (3.9) and (3.10) Equation (3.9) can be transformed to:

Definition 3.12 (Field Weight as Information Content Expanded)**.**

$$w_f(f,q,c) := \sum_{t \in q \cap f} -\log \frac{\mathrm{df}(t,f,c)}{\mathscr{N}(t,f,c)} \quad \left(\propto \sum_{t \in q \cap f} \mathrm{IDF}(t,F_i) \right) \tag{3.11}$$

It is obvious that Definition 3.12 is effectively the sum of field-based IDFs for the intersection of a query and a document field.

3.5.2 Model Candidates

We experiment with three different model candidates. More specifically, three definitions for $\mathscr{N}$ are compared to get an understanding of how to best estimate the number of documents in which term t could potentially occur , i.e. those that are not empty.

Definition 3.13 (Total Number of Document Fields $\mathscr{N}_{\mathrm{tot}}$)**.**

$$\mathscr{N}_{\mathrm{tot}}(c) := N(c) \tag{3.12}$$

$\mathscr{N}_{\mathrm{tot}}$ simply defines the number of documents where term t could occur as the total number of documents in the collection.

Definition 3.14 (Number of Non-empty Document Fields $\mathscr{N}_{\text{non-empty}}$)**.**

$$\mathscr{N}_{\text{non-empty}}(t,f,c) := |\{d | f \in F_c \wedge \exists t, f_d : n(t,f_d) > 0\}| \tag{3.13}$$

$\mathscr{N}_{\text{non-empty}}$ ensures that fields which are empty for many documents are given less weight than they would otherwise. This makes sense as often fields are empty for reasons, such as data redundancies.

Definition 3.15 (Norm Number of Non-empty Document Fields $\mathscr{N}_{\text{norm}}$).

$$\mathscr{N}_{\text{norm}}(t, f, c) := N_{\text{non-empty}} \frac{\text{avgfl}(c)}{\text{avgfl}(f)} \tag{3.14}$$

$\mathscr{N}_{\text{norm}}$ ensures that short fields get more weight. Adding weight to shorter fields has been shown to be beneficial in previous research [69].

3.5.3 Evaluation and Analysis

The aim of the following experimentation is to demonstrate the proposed BM25-FIC model. It serves as a starting point for the following two chapters: A first iteration of an information content-based field weighting retrieval model is tested on two very different data collection. Understanding the strengths and weaknesses of this first iteration helps guide the research in the following chapters.

The evaluation seeks to answer the following three research questions, which correspond to different degrees of optimisation of the model candidatates and baselines. As discussed in the introduction chapter, in InvIR scenarios there is usually little training data available. For this reason the main focus in on RQ1, where no optimization is performed.

RQ1: How does BM25-FIC compare to baseline models when no optimization is performed?

RQ2: How does BM25-FIC compare to baseline models when the underlying model (BM25) is optimised?

RQ3: How does BM25-FIC compare to baseline models where field weights have been optimised?

Data Collection

For evaluation, we consider two benchmark datasets: the Kaggle Home Depot product catalogue data set[3], also used by Balananesinkordan et al. [30] and DBpediaV2 by Hasibi et al. [94].

The Homedepot data set contains 55k products with name, description and attribute fields. The attribute field contains additional information, such as notes and can also be empty. We considered 1000 queries with the most relevance judgements available. The documents were judged by humans on a scale between 1 and 3. Altogether there are 12,093 judgements, 10,260 relevant and 1,833 non-relevant.

[3] https://www.kaggle.com/c/home-depot-product-search-relevance/data

DBpedia2 is the second variant of a test collection by Balog et al. [96] extended by Hasibi el al. [94]. The data consists of 4.6 million Wikipedia entries with the fields: *labels*, *categories*, *similar entities*, *related entities* and *attributes*. The 463 queries for DBpedia are divided into four categories: Named entity queries, IR-style keyword queries, natural language questions and list search queries. We consider all these categories together. See Hasibi et al. [94] for more details on the exact definition of the fields and more information on the collection in general.

Baselines

The baseline models considered for the experimentation are FSA-BM25 (2.29) and the simpler version of BM25F with no field level-based document length normalization [69]. In this section, we assume that there is no training data or semantic knowledge of the fields available. This means that the field weights for FSA-BM25 and BM25F have to be set as uniform. The BM25 hyperparameters have been set according to the usual recommendation as $b = 0.8$ and $k_1 = 1.6$, where $k_1 = 1.6$ is the mid point of the recommended $[1.2 - 1.8]$ range [19].

RQ1: How does BM25-FIC compare to baseline models when no optimization is performed?

Table 3.1 shows the performance of the baseline models, as well as the BM25-FIC candidates when no optimization is applied. We can see that there is a clear difference between the two datasets. First of all, for Homedepot FSA-BM25 is the better-performing baseline — though not by a large margin — whereas for DBpedia BM25F is significantly better. However, more importantly, the BM25-FIC candidates do remarkably well for Homedepot and remarkably bad for DBpedia. For Homedepot the increase in MAP and NDCG@100 is about

data collection	Homedepot		DBpedia	
Baselines				
metric	MAP	NDCG@100	MAP	NDCG@100
FSA-BM25	*0.246*	*0.444*	0.224	0.351
BM25FSimple	0.238	0.429	***0.285***	***0.433***
Model Candidates				
metric	MAP	NDCG@100	MAP	NDCG@100
BM25-FIC-Tot	0.259	0.459	0.204	0.326
BM25-FIC-Non	0.263	0.464	0.207	0.331
BM25-FIC-Norm	**0.290**	**0.492**	0.194	0.314

Table 3.1: Results for Homedepot

0.05 in absolute terms and for DBpedia, the decrease is around 0.1 in absolute terms for both MAP and NDCG@100[4].

Both FSA-BM25 and BM25-FIC use FSA as the underlying aggregation method. It would seem from the results that where FSA does well, so does BM25-FIC. In order to understand better why the performance of FSA-BM25 was so bad for DBpedia two questions were posed: 1. was FSA-BM25 better for some queries, as was the case for Homedepot, or was the poor performance purely due to the data collection and 2. what could be possible reasons for such a big difference in the performance of FSA-BM25 and BM25-FIC between Homedepot and DBpedia

Answering the first question is easy, there are indeed many queries (30+) where FSA-BM25 does significantly better than BM25F for DBpedia, however, there are significantly more queries where the BM25F does better than FSA-BM25. Answering the second one is more difficult. To give some hints as to the reason for one model doing better than the other for different queries, the queries with the largest difference in performances between the models were analysed. Table 3.2 presents these queries.

	BM25F	FSA-BM25
1	concord steel	In which city was the former Dutch queen Juliana buried?
2	banana paper making	Campuses of Indiana University
3	daggeroso inclined to use a dagger novel Sons and Lovers	st paul saints
4	Paul Auster novels	shobana masala
5	chase masterson	Scott Counti
6	What did Bruce Carver die from?	charles darwin
7	ashley wagner	the morning call lehigh valley pa
8	rock 103 memphi	birds cannot fly
9	the dish danielle fishel	In which U.S. state is Area 51 located?
10	Which books by Kerouac were published by Viking Press?	bradley center

Table 3.2: Top 10 queries for each model in terms of $\Delta\,\mathrm{MAP}$. BM25F-column: $\Delta\,\mathrm{MAP} = \mathrm{MAP}_{\mathrm{BM25F}} - \mathrm{MAP}_{\text{FSA-BM25}}$ and FSA-BM25-column: $\Delta\,\mathrm{MAP} = \mathrm{MAP}_{\text{FSA-BM25}} - \mathrm{MAP}_{\mathrm{BM25F}}$

The reasons why BM25F does better than FSA-based models have been analysed extensively by Robertson et al. [69]. As discussed in Section 2.6, saturating BM25F saturates term frequencies across fields, meaning term occurrences in different fields are not "double counted". Take the query "concord steel": The

[4] Percentage changes not reported as percentages of arithmetic means are non-sensible

relevant document article is “Shougang Concord International”, a Chinese steel company. It is obvious that for this query it is paramount that both the query terms appear in some of the document fields, otherwise things like “concord planes”, get high scores. Effectively, any document that does not have both the query terms is immediately non-relevant. As FSA-BM25 does consider term occurrences across fields, documents with the term “concord” found in two fields will receive a higher score than those with the term “concord” in one field and “steel” in another. It is because the BM25F gives more weight to the occurrence of steel in other fields, rather than further occurrences of “concord” (concord has a higher IDF), that it does better.

However, as mentioned for some queries in the DBpedia benchmark, FSA-BM25 does much better than BM25F. Furthermore, for Homedepot overall, the performance of FSA-based models, especially BM25-FIC does significantly better. Some possible reasons for this are suggested by the FSA-BM25 column in Table 3.2: The clearest example of a query FSA-BM25 does well on is “Campuses of Indiana University”. Firstly, we would expect relevant documents to contain a field with all three query terms (“of” dropped as a stopword). This is because we are not interested in all Universities in Indiana, just Indiana University campuses. Secondly, the model benefits from considering the fields separately: For relevant documents the term “University” occurs in all five fields, having slightly different meanings in each. FSA-BM25 gives more weight to such documents, whereas BM25F would give the same importance to three occurrences of “University” in the *attributes* field as it would for three occurrences of “University” spread over three different fields.

The above discussion regarding the queries in Table 3.1 should not be seen as a formal analysis of the models and their behaviour. Theorizing based on individual queries does not produce conclusions that can be generalized. Instead, the discussion should be read as the underlying motivation to examine the performance and behaviour of both TFA and FSA-based models more closely, in order to understand whether it is possible to develop a model which has both of their strengths. The next Chapter is where we formally analyse and examine these things more closely.

How does BM25-FIC compare to baseline models when the underlying model (BM25) is optimised?

Table 3.3 shows the results of the experimentation when the underlying model hyperparameters k_1 and b have been optimised. The optimisation was performed using coordinate ascent (CA), optimizing for NDCG with 5-fold cross validation [106].

data collection	Homedepot		DBpedia	
Baselines				
metric	map	ndcg@100	map	ndcg@100
FSA-BM25	*0.314*	*0.510*	0.284	0.434
BM25FSimple	0.243	0.433	*0.308*	*0.463*
Model Candidates				
metric	map	ndcg@100	map	ndcg@100
BM25-FIC-Tot	0.343	0.534	0.323	0.473
BM25-FIC-Non	**0.344**	**0.535**	**0.323**	**0.474**
BM25-FIC-Norm	0.314	0.510	0.281	0.430

Table 3.3: Results for Homedepot

As expected, optimising the k_1 and b parameters for the candidate models and the baseline increases accuracy in all cases. The trends within the baselines stay as they were in the previous section. However, there is an important difference in the results for the DBpedia dataset in terms of the baseline results compared to our candidate models. As opposed to completely non-optimised scenarios, BM25-FIC-Tot and BM25-FIC-Non outperform both baselines. This is highly interesting as it suggests that optimising the parameters compensates for the fact that BM25-FIC is an FSA-based model and thus term frequency is not saturated across the fields. A potential explanation for this is the k_1 hyperparameter specifically. When optimised k_1 is just above 0, or even 0 depending on the field. This suggests that by considering within-field term frequencies higher than one as one, reduces the noise created by a lack of cross-field term frequency saturation to the extent that an FSA-based model can outperform a TFA-based in some circumstances. Of course, this should not be considered good model behaviour, as the model basically dismisses within field term frequencies altogether. Term frequency is an important part of any retrieval model and thus should be taken into account, meaning k_1 should not be set to 0, except in very special circumstances. How to solve the issue of FSA models lacking cross-field term frequency saturation, without having to dismiss term frequencies higher than 1 will be one of the main problems to be solved throughout the rest of the thesis.

How does BM25-FIC compare to baseline models where field weights have been optimised?

Table 3.4 shows the results of the experimentation for the case where the field weights and hyperparameters for the baseline models have been optimised. For the candidate models, only the hyperparameters k_1 and b have been optimised. The candidate models do not have field weights to optimise, so the main idea of this exercise is to understand how much worse they performed compared to baselines where field weights are optimised.

data collection	Homedepot		DBpedia	
	Baselines			
metric	map	ndcg@100	map	ndcg@100
FSA-BM25	***0.352***	***0.538***	0.317	0.473
BM25FSimple	0.337	0.526	***0.330***	***0.483***
	Model Candidates			
metric	map	ndcg@100	map	ndcg@100
BM25-FIC-Tot	0.343	0.534	0.323	0.473
BM25-FIC-Non	0.344	0.535	0.323	0.474
BM25-FIC-Norm	0.314	0.510	0.281	0.430

Table 3.4: Results for Homedepot

From Table 3.4 we can see that the performance difference between the fully optimised baselines and the candidate models is smaller than we would perhaps expect. For example, for Homedepot the FSA-BM25 only barely beats the BM25-FIC-Tot and BM25-FIC-Non candidate models, whilst BM25FSimple in fact does worse than BM25-FIC-Tot and BM25-FIC-Non. For DBpedia the same is true except the better-performing baseline is BM25FSimple.

What these results suggest is that the BM25-FIC model and the underlying information content-based field weighting are able to leverage the structure of the document in similar ways as the FSA-BM25 and BM25F models when their field weights are optimised. Demonstrating that this is indeed the case and formulating a more advanced iteration of the BM25-FIC model will be the topic of Chapter 5. However, first, the next chapter will establish a more concrete and robust framework for discussing and analysing how within-field term frequencies are modelled in SDR. This will be done through the formulation of formal constraints for SDR.

3.6 Summary, Conclusions and Contributions

This chapter covered the following issues:

- Intuitive justification for the use of information content for field weighting. Borrowing the concepts of exhaustivity and specificity, as well as the work Spark-Jones from 50 years ago, the intuition for the use of information content for field weighting was defined as an attempt to define the specificity of a field as a statistical, rather than a semantic property of the field.

- Theoretical justification for the use of information content for field weighting. Two justifications offered: 1. by mirroring the justification for the IDF by Aizawa [103], the use of information content is justified using information theory. 2. By borrowing a more flexible definition of information content by Hintikka [40], its use is justified in a similar manner as it is in the DFR model.
- Discussion on the background and context using information content-based field weighting, focusing on how it relates to existing atomic and structured retrieval models.
- A study on a simple field weighting method for the BM25. Experiments were performed on two datasets. Initial small-scale results guide the next chapter.

The main conclusions are:

- The use of information content for field weighting can be justified — to an extent — both intuitively and theoretically.
- A simple version of the approach (BM25-FIC) works very well on the Homedepot collection and very badly on the DBpedia collection.
- The underlying reasons for the above discrepancy would seem to have something to do with how the model manages to appreciate occurrences of terms in multiple fields, but fails to saturate term frequency across fields. However, at this point, these observations are very vague and cannot be backed by evidence.

The main contributions are:

- Introduction of a new approach to field weighting in SDR with a well-founded conceptual/intuitive and theoretical framework.
- A study using a simple method for information content-based field weighting (BM25-FIC) with formal evaluation.

Chapter 4

Formal Constraints for Structured Document Retrieval (SDR)

This chapter introduces formal retrieval constraints for SDR. Large parts of its content were published in ICTIR'22 [2]. The chapter is structured as follows:

- Section 4.1 briefly describes the motivation behind formulating constraints for SDR in the context of this thesis and in general.
- Section 4.2 introduces the proposed approach and the chapter's contributions in the context of SDR in general.
- Section 4.3 formally describes the constraints for SDR.
- Section 4.4 analyses how existing models satisfy and fail to satisfy the proposed constraints.
- Section 4.5 details the experimentation and analysis demonstrating how the constraints affect the ranking behaviour of various models on benchmark data collections.
- Section 4.6 concludes.

4.1 Motivation

Chapter 3 demonstrated that BM25-FIC worked well for the Homedepot dataset, but not as well for the DBpedia dataset. Specifically, for the non-optimised task, BM25-FIC performed very well for Homedepot and very badly for DBpedia. The experimentation suggested that one of the main reasons for poor performance on DBpedia was the lack of term frequency saturation across fields, an observation also made by Robertson et al. [69]. However, BM25-FIC and FSA-BM25 outperformed BM25F for the Homedepot data, even though BM25F does saturate term frequency across fields. Closer inspection suggested that this might be because the FSA-based models appreciate terms occurring in multiple fields more than them occurring in just one. So it seemed like there might be a trade-off between these two models (FSA vs. TFA) on different data collections. However, trying to identify which approach is better for what data collection would be at this point heuristic at best. Therefore, a formal analysis of the trade-off was required.

Fang et al. [26] introduced their (atomic) retrieval constraints in order to "capture retrieval heuristics, such as the TF-IDF, in a formal way, making it possible to apply them to any retrieval formula analytically." This chapter does something similar, except it is not heuristics we capture (what they consider heuristics has been formalized in many cases [92, 39, 103, 29]), but intuitive "rules", that SDR models should aim to follow. By using these constraints hopefully, we can better understand exactly how the different aggregation functions (FSA and TFA) behave in terms of ranking and how we could further develop information content-based field weighting to work across different retrieval scenarios and data collections.

4.2 Introduction

Analytical retrieval models, such as the BM25 and Language Modelling (LM), are used widely in commercial and academic settings. The behaviour of these models is understood well due to extensive research over the last 20+ years. One important line of enquiry has been formal retrieval constraints / axioms [26, 27]. The aim of this chapter is to develop such a framework for structured document retrieval (SDR). This is accomplished by identifying four constraints that define optimal ranking behaviour in simple, but informative scenarios, by analysing how existing models adhere to those constraints and by testing how satisfying the constraints affects retrieval behaviour.

Table 4.1 summarises the intuition underlying the four chosen constraints for SDR. There are of course many more possible constraints. These four constraints have been chosen because they lead to intuitive ranking behaviour by avoiding some issues structured data creates. The documents in Table 4.2 will be used to demonstrate these issues. A hypothetical model satisfying all four constraints would rank the documents as RSV(d1) > RSV(d2) > RSV(d3) > RSV(d4) > RSV(d5). This results from the fact that intuitively d1 should be ranked first

Constraint	Abbr.	Intuition
Term distinctiveness	TD-Co	Adding unseen query terms to a document should increase the retrieval score more than adding query terms already considered
Field distinctiveness	FD-Co	Adding a query term to a new field should increase the retrieval score more than adding it to a field where it already occurs
Term importance	TI-Co	A model should consider the importance of a term on a field level, rather than document-level
Field importance	FI-Co	A model should be able to boost or decrease the weight given to a field, based on some notion of field importance

Table 4.1: Intuition underlying formal constraints for SDR. Field refers to a field of a document; e.g. *abstract* or *author.*

because it contains both query terms, d2 should be second because the one query term appears twice and in different fields, d3 should be third because the one query term occurs twice in the same field, d4 should be fourth as it only contains one occurrence of a query term and d5 should be last because it also only contains one occurrence, but in a field where the term has a lower IDF.

field	plot		description		flattened doc
term	english	spy	english	spy	
field-specific IDF	1.9	2.5	2.0	2.1	
document 1	1	0	0	1	english spy
document 2	1	0	1	0	english english
document 3	0	0	2	0	english english
document 4	0	1	0	0	spy
document 5	0	0	0	1	spy

Table 4.2: Example with two fields and two query terms illustrating how rankings by existing SDR models are not always intuitive.

The **first contribution** of this chapter is to formalize retrieval constraints that guarantee this ranking. It is not the intention of this chapter to claim that the described ranking behaviour is always the correct one, as this is defined by the user. Instead, the intention is to formulate constraints that produce an intuitive ranking where no knowledge of user preferences is available. The lack of knowledge about user preferences also means that field weights are set as uniform.

The **second contribution** is to analyse why and how widely used SDR models satisfy, or fail to satisfy the constraints and how this affects their ranking performance on benchmark datasets. It will be shown that the underlying reasons have to do with how they model term frequency across different fields and how they model document structure in general. FSA-based models consider term frequency to be independent across fields, an assumption which was shown to be harmful by Robertson el al. amongst others [69, 92] and results in the TD-constraint not being satisfied ($\mathrm{RSV}(d_1) = \mathrm{RSV}(d_2)$ rather than $\mathrm{RSV}(d_1) > \mathrm{RSV}(d_2)$ in Table 4.2). On the other hand, TFA-based models, such as BM25F and Mixture of Language Models (MLM), consider the document as atomic (rather than structured) after term frequency weighting, meaning they fail to fulfil the FD-constraint ($\mathrm{RSV}(d_2) = \mathrm{RSV}(d_3)$ rather than $\mathrm{RSV}(d_2) > \mathrm{RSV}(d_3)$ in Table 4.2). Since TFA-based models sum the term frequencies together before considering their importance or specificity (e.g. IDF), they also fail to consider the Term Importance constraint ($\mathrm{RSV}(d_4) = \mathrm{RSV}(d_5)$ rather than $\mathrm{RSV}(d_4) > \mathrm{RSV}(d_5)$ in Table 4.2). There are models such as PRMS that fail to consider field importance in any way, thus failing to satisfy the Field Importance constraint and often the Term Importance Constraint as well.

The **third contribution** is to discuss how SDR models could be developed in the future to better satisfy the constraints. Our findings suggest that in order for an SDR model to accomplish this, it should be able to balance between saturating term frequency across fields, whilst still explicitly considering the document structure. The next chapter further develops the use of information content-based field weighting to do just this.

4.3 Constraints for Structured Retrieval

Regarding the example in Table 4.2, the following constraints lead to an intuitive ranking: As mentioned above, d1 should be ranked first because it contains both query terms, d2 should be second because the one query term appears twice and in different fields, d3 should be third because the one query term occurs twice in the same field, d4 should be fourth as it only contains one occurrence of a query term and d5 should be last because it also only contains one occurrence, but in a field where the term has a lower IDF. It is worth noting that the use of IDF in Table 4.2 refers to the use specificity of a term in general, understood as its discriminative power, or information content. IDF is used for clarity, as the majority of analysis in this chapter focuses on the BM25.

This “intuitive ranking” does not necessarily represent the “correct ranking”, as this is ultimately judged by the user. For example, the user might be more interested in the *description* field, in which case it might make sense to rank d3 higher than d2. However, lacking this kind of knowledge of user preferences, the ranking behaviour described above does correspond to four intuitive rules:

1. With all else equal, documents with many distinct query terms should rank higher than those with few (Term Distinctiveness)
2. With all else equal, documents where a query term occurs in several fields should rank higher than if the term occurs only in a few fields (Field Distinctiveness).
3. With all else equal, documents where a query term occurs in a field where it is rare, should rank higher than documents where it occurs in a field where it is common (Term Importance).
4. With all else equal, documents where a query term occurs in a field that is important should rank higher than documents where the term occurs in a less important field (Field Importance)

Transforming these rules into formal retrieval constraints is done in the following subsections and is the main contribution of this chapter.

4.3.1 Term Distinctiveness: TD-Co

Definition 4.1 (Term Distinctiveness (TD-Co))**.** *Let Q denote a query, S a retrieval score and d a document. A document has a set of m fields: $\{f_1 \dots f_m\}$. Here the field f in which term t_i occurs is irrelevant so an occurrence of term t_i is denoted as t_i. Let T_d be a set of query terms that occur in document d.*

$$\forall Q, d, f, t: \ \textit{if } t_k \notin T_d \textit{ and } t_j \in T_d \textit{ then } S(Q, d \cup t_k) > S(Q, d \cup t_j) \tag{4.1}$$

I.e. adding many distinct query terms to a document should increase the score more than adding a few, no matter in which fields they appear. For the documents in Table 4.2 this would mean that document d1 ranks higher than d2. The satisfaction of this constraint is central to the BM25F retrieval model. By saturating term frequency across fields, the BM25F gives more importance to the first occurrence of a query term, compared to subsequent occurrences of the same term, wherever in the document they occur [69]. By doing so, it puts more emphasis on a document having many distinct query terms, rather than a few. This logic is one of the central aspects of the BM25F, which has been shown to outperform FSA-based models for various data collections [69, 70].

In essence, the TD-Constraints is communicating a similar issue as constraint TFC3 by Fang et al. in [26]. However, it is worth re-formalizing it for SDR because, 1. it will be shown that many SDR models do not satisfy it, whereas in atomic retrieval this is not common and 2. its implications are more severe for SDR, as term frequencies are often inflated through field weights. Furthermore, instead of defining the constraint only for scenarios where the IDF values of terms are equal, we analyse the satisfaction of the constraint for a more general case.

4.3.2 Field Distinctiveness: FD-Co

Definition 4.2 (Field Distinctiveness (FD-Co))**.** *Let Q denote a query, S a retrieval score and d a document. A document has a set of m fields: $\{f_1 \dots f_m\}$. The occurrence of term t_i in document field f is denoted $t_{i,f}$, meaning a document is modelled as a set of term occurrences over a set of fields: $d = \{t_{a,f1}, t_{a,f2}, \dots, t_{b,f1}, \dots\}$. Let $F_d(t)$ denote a set of fields f in document d with an occurrence of term t.*

$$\forall Q, d, f, t: \textit{ if } f_k \notin F_d(t) \textit{ and } f_j \in F_d(t) \textit{ then } S(Q, d \cup t_{i,fk}) > S(Q, d \cup t_{i,fj}) \tag{4.2}$$

In other words, the more fields a query term appears in, the higher the ranking score of the document should be. This constraint also implies that adding a query term to a new field of a document should increase the ranking score more than adding a query term to the field where it already appears. For the documents in Table 4.2, this would mean that document d2 ranks higher than d3.

The order of $\{t_{a,f1} \dots t_{a,fm}\}$ does not refer to the order of the fields in the documents, but the order in which query term t_a occurs in them, meaning f_1 is not the first field of the document, but the first field in which t_a occurs.

4.3.3 Term Importance: TI-Co

Definition 4.3 (Term Importance (TI-Co))**.** *Let Q denote a query, S a retrieval score and d a document. A document has a set of m fields: $\{f_1 \dots f_m\}$. $I(t)$ denotes the importance of a term (e.g. IDF(t)).*

$$\forall Q, d, f, t: \textit{ if } I(t_k) > I(t_j) \textit{ then } S(Q, d \cup t_k) > S(Q, d \cup t_j) \tag{4.3}$$

The underlying idea behind the **TI-constraint** is that a term might carry a different meaning depending on the field it occurs in, and therefore its occurrences in different fields should be treated separately. In terms of Table 4.2, the TI-constraint concerns the ranking of the last two documents. If it is not satisfied, the same weight is given to the occurrence of "spy" in the *plot* field and the *description* field, even though the field-specific IDF values are different. Such a model effectively sees documents d4 and d5 as the same, leading to them being ranked in a non-intuitive manner.

4.3.4 Field Importance: FI-Co

Definition 4.4 (Field Importance (FI-Co)). *Let Q denote a query, S a retrieval score and d a document. A document has a set of m fields: $\{f_1 \dots f_m\}$. $I(f_i)$ denotes the importance of field f_i.*

$$\forall Q, d, f, t: \text{ if } I(f_k) > I(f_j) \text{ then } S(Q, d \cup t_{i,fk}) > S(Q, d \cup t_{i,fj}) \tag{4.4}$$

In other words, adding a query term to a field with greater importance must increase the score more than adding one to a field with lower importance. For the documents in Table 4.2, this would mean that document 1 would rank higher than document 2 if the *plot* field were boosted due to some knowledge of its importance. This might seem trivial, but the point being made is that an SDR model should be able to weight fields based on some notion of importance. This weighting can be done through learning field weights or using heuristics for example.

4.4 Constraint Satisfaction by Existing Models

	Aggr.	Term Dist. TD-Co	Field Dist. FD-Co	Term Imp. TI-Co	Field Imp. FI-Co
PRMS	FSA	NO	Cond.	NO	YES
FSA	FSA	NO	Cond.	YES	YES
BM25-FIC	FSA	NO	Cond.	YES	YES
BM25F	TFA	Cond.	NO	NO	YES
MLM	TFA	Cond.	NO	YES	YES
FSDM	TFA	Cond.	NO	NO	YES

Table 4.3: Constraint satisfaction of SDR models: Cond. = Conditional. Conditional means that collection statistics need to be considered.

Table 4.3 shows which SDR model satisfies which constraints. Conditional satisfaction of a constraint refers to cases where collection statistics need to be accounted for, i.e. the specificity / IDF of query terms for example. Whereas Fang et al. assume the IDFs of query terms to be equal, the analysis in this chapter looks at what levels of specificities cause a model to satisfy, or not satisfy a given constraint. In simple terms, if we were to assume that IDFs are equal for all terms, the "conditional" entries in Table 4.3 could be changed to "YES".

4.4.1 Term Distinctiveness Satisfaction

FSA models do not satisfy TD Constraint. This is because the field-based scores are summed together, with no regard to whether both query terms (*english* AND *spy*) occur, or only one of them. The issue is evident from Table 4.2. Assuming equal specificity weights (e.g. IDF), d_1 and d_2 are rank equal. Intuitively we want documents with more query terms to rank higher. The problem comes from the fact that FSA assumes term frequency to be independent across fields for a given term, thus "double accounting" the occurrence of *english.* TFA solves this by saturating term frequencies across the fields, i.e. it assumes a constant dependency of term occurrences between fields for a given term. It has been shown that this significantly increases the robustness of the models and makes them less noisy [69, 70].

The satisfaction of the TD-Constraint is conditional for the TFA-based models. They suffer from the same issue as atomic models when it comes to the specificity ratio of query terms as discussed in Chapter 2. There exists a threshold for the ratio of IDF values between query terms at which a second occurrence of a query term can dominate over the first occurrence of another query term.

The following will explain this conditionality in the general case for BM25F, after which we will discuss how the general case can be simplified to capture the conditionality of satisfying TD-Constraint in a more intuitive way.

Definition 4.5 (Cross-Term IDF Ratio)**.** *The ratio of the IDF values between terms b and a is denoted* IDF-CT-Rat(t_a, t_b, c)*. Let t be a term and c a collection.*

$$\text{IDF-CT-Rat}(t_a, t_b, c) := \frac{\text{IDF}(t_b, c)}{\text{IDF}(t_a, c)} \tag{4.5}$$

Definition 4.6 (Cross-Term IDF Ratio Threshold)**.** *Let $q = \{t_1, \ldots, t_n\}$ be a query, d a document with T occurrences of term t_a in field f_i and z occurrences of term t_b in another field f_j. Let $\overline{d}$ be an amended version of d, where the occurrences of term t_b in f_j have been replaced by occurrences of t_a.*

$$\text{IDF-CT-Rat}_{\text{th}}(t_a, t_b, c, k_1) := \frac{\frac{w_i T + w_j z}{k_1 + w_i T + w_j z} - \frac{w_i T}{k_1 + w_i T}}{\frac{w_j z}{k_1 + w_j z}} \tag{4.6}$$

IDF-CT-Rat$_{\text{th}}(t_a, t_b, c, k_1)$ defines the threshold for IDF-CT-Rat(t_a, t_b, c) above which score$(d) >$ score$(\overline{d})$. It is worth noting that strictly speaking, the TD-Constraint states that $T = z = 1$. However, Definition 4.6 and Theorem 4.1 do not make this assumption. Instead, they solve the problem for a general case, which we will then simplify. Formal theorem and proof are below.

The underlying idea of the cross-term IDF ratio theorem is that there exists a threshold for IDF-CT-Rat(t_a, t_b, c) below which TD-Constraint is not satisfied by BM25F, meaning the documents d_1 and d_2 in Table 4.2 would be ranked incorrectly.

Theorem 4.1 (BM25F and the Term Distinct. Constraint)**.** *Let $q = \{t_1, \ldots, t_n\}$ be a query, d a document with T occurrences of term t_a in* *field f_i and z occurrences of term t_b in another field f_j. Let $\overline{d}$ be an amended version of d, where the occurrences of term t_b in f_j have been replaced by occurrences of t_a.*

$$\begin{aligned}
&\forall(t_a, t_b) \in q \cap d : \\
&\qquad \text{IDF-CT-Rat}(t_a, t_b, c) > \text{IDF-CT-Rat}_{\text{th}}(t_a, t_b, c, k_1) \\
&\qquad \Rightarrow \text{RSV}_{\text{BM25F}}(q, d, c) > \text{RSV}_{\text{BM25F}}(q, \overline{d}, c) \qquad (4.7)
\end{aligned}$$

Proof. Following Definition 4.5 the threshold for satisfying the TD-Constraint becomes

$$\frac{\text{IDF}(t_b, c)}{\text{IDF}(t_a, c)} > \frac{\frac{w_iT+w_jz}{k_1+w_iT+w_jz} - \frac{w_iT}{k_1+w_iT}}{\frac{w_jz}{k_1+w_jz}} \tag{4.8}$$

$$\begin{aligned}
&\frac{w_iT}{k_1 + w_iT}\,\text{IDF}(t_a, c) + \frac{w_jz}{k_1 + w_jz}\,\text{IDF}(t_b, c) \\
&> \frac{w_iT + w_jz}{k_1 + w_iT + w_jz}\,\text{IDF}(t_a, c) \qquad (4.9)
\end{aligned}$$

Since $|d| = |\overline{d}|$, the ranking of the documents, i.e. the inequality of the scores is not affected by document length normalisation. Therefore we can set $n_{\text{norm}}(t, f, d) = n(t, f, d)$ in Eqn. (2.52) without changing the analysis. Assuming the term frequencies from the theorem and following Eqn. (2.52) we can re-write Eqn. (4.9) as

$$\text{RSV}_{\text{BM25F},k_1,b}(q, d, c) > \text{RSV}_{\text{BM25F},k_1,b}(q, \overline{d}, c) \tag{4.10}$$

□

In order to understand how BM25F satisfies the constraints more intuitively and in terms of Table 4.2, we assume uniform field weights and set $T = z = 1$ (see documents d1 and d2 in the example). This simplifies Eqn 4.6 to:

$$\text{IDF-CT-Rat}_{\text{th}}(t_a, t_b, c, k_1) = \frac{2k_1 + 2}{k_1 + 2} - 1 \tag{4.11}$$

Eqn. (4.11) shows that whether the BM25F satisfies the TD-Constraint depends on the ratio of the IDFs and the term frequency saturation parameter k_1. If $k_1 = 2.0$ the ratio of IDF values below which BM25F would fail to satisfy the TD-Constraint equals 0.5. So if the rarest term of the query has an IDF twice the size of the most common term, the constraint is not satisfied. There are cases where it makes sense for a model to not satisfy the TD-Constraint, for example, if the common term is a stopword. The IDF value for stopwords tends to be very close to 0, so the constraint is obviously not satisfied, nor should it be.

However, a term can easily have half the IDF of another and still be important, so the conditionality of the TD-Constraint should be considered analytically. This issue is present in both SDR and atomic retrieval. The following discusses how it might be more severe for SDR, due to field weighting.

Consider a scenario where $k_1 = 2.0$ and the occurrences of t_a for $\overline{d}$ in f_j occur in the third field f_k. The field weights are $w_{f_i} = 1$, $w_{f_j} = 1$ and $w_k = 3$ Maybe f_k is a *title* of the book and we wish to boost it compared to the *abstract* and *body* for example. In such a situation an occurrence of the new term t_b in field f_j would have the same effect on the score, as a second occurrence of t_a in f_k, even if $\text{IDF}(t_a) = \text{IDF}(t_b)$, i.e. the TD-Constraint would not be satisfied even if the terms had the same IDF. The key takeaway here is that when heuristically boosting fields because they are important — say the title of a book — other hyperparameters should be considered as well. In order for the field boosting to work, it is therefore likely that all the parameters have to be optimised using supervised learning of some form. In this instance k_1 would have to be adjusted to set the degree of term frequency saturation, other SDR models such as the MLM are not able to adjust this degree well, which might lead to worse performance.

4.4.2 Field Distinctiveness Satisfaction

TFA models (MLM, BM25F, FSDM) do not satisfy the FD-Constraint. After applying the field weights at the term level, they consider the document as atomic. This issue is obvious in the retrieval scenario in Table 4.2 for the ranking of documents d2 and d3: Assuming equal IDF values, it does not matter whether *english* appears twice in *description*, or once in *plot* AND once *description*, the documents get the same rank-score.

Satisfying the FD-Constraint is conditional for the FSA-based models. The following will explain this conditionality in the general case for FSA-BM25, after which we will discuss how the general case can be simplified to capture the conditionality of satisfying the FD-Constraint in a more intuitive way.

Definition 4.7 (Cross-Field IDF Ratio)**.** *The ratio of the IDF-values between fields j and i for term t is denoted* $\text{IDF-CF-Rat}(t, F_j, F_i)$.

$$\text{IDF-CF-Rat}(t, F_j, F_i) := \frac{\text{IDF}(t, F_j)}{\text{IDF}(t, F_i)} \tag{4.12}$$

Definition 4.8 (Cross-Field IDF Ratio Threshold)**.** *Let $q = \{t_1, \ldots, t_n\}$ be a query, d a document with T occurrences of term t in field f_i and z occurrences of term t in another field f_j. Let $\overline{d}$ be an amended version of d, where the occurrences of term t in f_j have been moved to f_i and z occurrences of non-*

query terms have removed from f_i and added to f_j. These non-query terms ensure that IDF-CF-Rat *is only concerned with query term occurrences, rather than document lengths.*

$$\text{IDF-CF-Rat}_{\text{th}}(t, F_i, F_j, k_1) := \frac{w_i}{w_j} \frac{\frac{T+z}{k_1+T+z} - \frac{T}{T+k_1}}{\frac{z}{z+k_1}} \tag{4.13}$$

IDF-CF-Rat$_{\text{th}}(t, F_i, F_j, k_1)$ defines the threshold for IDF-CF-Rat(t, F_j, F_i) above which the FD-Constraint is satisfied, meaning $\text{RSV}_{\text{FSA,M}}(d, q, c) > \text{RSV}_{\text{FSA,M}}(\overline{d}, q, c)$. It is worth noting that strictly speaking, the FD-Constraint states that $T = z = 1$. However, Definition 4.6 and Theorem 4.1 do not make this assumption. Instead, they solve the problem for a general case, which we will then simplify. The formal theorem and proof are below.

The underlying idea of the cross-field IDF ratio theorem is that there exists a threshold for IDF-CF-Rat(t, F_i, F_j) below which FD-Constraint is not satisfied by FSA-BM25, meaning the documents d_2 and d_3 in Table 4.2 would be ranked incorrectly.

Theorem 4.2 (FSA and the Field Distinctiveness Constraint)**.** *Let $q = \{t_1, \ldots, t_n\}$ be a query, d a document with T occurrences of term t in field f_i and z occurrences of term t in another field f_j. Let $\overline{d}$ be an amended version of d, where the occurrences of term t in f_j have been moved to f_i and z occurrences of non-query terms have removed from f_i and added to f_j.*

$$\begin{aligned} &\forall t \text{ and } (F_i, F_j) \in q \cap d : \\ &\quad \text{IDF-CF-Rat}(t, F_i, F_j, k_1) > \text{IDF-CF-Rat}_{\text{th}}(t, F_i, F_j, k_1) \\ &\quad \Rightarrow \text{RSV}_{\text{FSA,M}}(q, d, c) > \text{RSV}_{\text{FSA,M}}(q, \overline{d}, c) \end{aligned} \tag{4.14}$$

Proof. Following Definition 4.7 the threshold for satisfying the FD-Constraint becomes

$$\frac{\text{IDF}(t, F_j)}{\text{IDF}(t, F_i)} > \frac{w_i}{w_j} \frac{\frac{T+z}{k_1+T+z} - \frac{T}{T+k_1}}{\frac{z}{z+k_1}} \tag{4.15}$$

$$\begin{aligned} &w_i \frac{T}{T+k_1} \text{IDF}(t, F_i) + w_j \frac{z}{z+k_1} \text{IDF}(t, F_j) > \\ &w_i \frac{T+z}{k_1+T+z} \text{IDF}(t, F_i) \end{aligned} \tag{4.16}$$

The BM25 retrieval status value of field f is calculated as

$$\mathrm{RSV}_{\mathrm{BM25},k_1,b}(q, f, F) := \sum_{t \in q} \frac{n_{\mathrm{norm}}(t, f, b_f)}{k_1 + n_{\mathrm{norm}}(t, f, b_f)} \, \mathrm{IDF}(t, F) \tag{4.17}$$

Since $|d| = |\overline{d}|$, the ranking of the documents, i.e. the inequality of the scores is not affected by document length normalisation. Therefore we can set $n_{\mathrm{norm}}(t, f, d) = n(t, f, d)$ in Eqn. (4.17) without changing the analysis. Assuming the term frequencies from the theorem, and following Eqn. (4.17) we can re-write Eqn. (4.16) as

$$\mathrm{RSV}_{\mathrm{FSA,M}}(q, d, c) > \mathrm{RSV}_{\mathrm{FSA,M}}(q, \overline{d}, c) \tag{4.18}$$

□

In order to understand how FSA-BM25 satisfies the constraints more intuitively and in terms of Table 4.2, we assume uniform field weights and set $T = z = 1$ (see documents d2 and d3 in the example). This simplifies Eqn. (4.13) to:

$$\text{IDF-CF-Rat}_{\mathrm{th}}(t, F_i, F_j, k_1) = \frac{2k_1 + 2}{2 + k_1} - 1 \tag{4.19}$$

Meaning the FD-Constraint is satisfied by the FSA models as long as the ratio of the highest and lowest IDF-value for all terms is greater than $\frac{2k+2}{2+k_1} - 1$. This would be likely if the two fields are correlated in their content, as we would expect similar IDFs for a given term in both fields. The above analysis has focused on the BM25, however, FSA models can be used with any retrieval function. A similar analysis on LM would focus on the hyperparameter μ and the background model.

4.4.3 Term Importance Satisfaction

If the TI-Constraint is not satisfied the same weight is given to the occurrence of “spy” in the *plot* field and the *description* field, even though the field-specific IDF-values are different. The model effectively sees documents d4 and d5 as the same, leading to them being ranked in a non-intuitive manner. FSA models satisfy the TI-constraint as they consider each field separately with respect to term specificity. BM25F does not satisfy the TI-Constraint, as the document is flattened before the rarity of terms is considered. For MLM the constraint is satisfied as the background model enters Definition 2.32 at the field level.

4.4.4 Field Importance Satisfaction

The FI-constraint is the easiest to satisfy. As long as the model is able to give weight to fields based on their importance this constraint is satisfied. For FSA, BM25F, MLM and FSDM this can be done through field weighting. However, for PRMS this is not possible as the weight is not based on the importance of a field, but on how each query term is mapped to it.

4.5 Evaluation and Analysis

The following experimentation demonstrates how each of the proposed constraints affects ranking performance on established benchmark collections.

4.5.1 Data collections and Baselines

The main data collection for the experimentation is DBpedia [94]. The collection consists of 4.6m documents and 467 queries, which are divided into four query types: named entity queries (NEQ), IR-style keyword queries (KEY), natural language questions (NLQ) and list queries (LQ). There are 5 document fields: entity name, attributes (wiki page full info), categories, similar entities and related entities. For more details see [94]. In order to analyse the TI-Constraint, single-term queries are required, of which there are few for DBpedia. For this part of the analysis, we use the Homedepot dataset[1]. There are 55k documents and 10k+ queries, some of which have very few relevance judgements. We have chosen the 1000 queries with the most judgements and out of those only consider the ones with a single query term (n=65). BM25 is used as the underlying model for the analysis as there exists a strong precedent for its use in both atomic and structured retrieval research.

4.5.2 Field and Term Distinctiveness: What is their Relative Importance?

So far it has been demonstrated that there is a trade-off between the FD and TD-constraints for FSA and TFA. To analyse this trade-off, we compare the performance of **BM25F** and **FSA-BM25**. For FSA-BM25 we use the original version of the BM25 by Robertson el al. [33, 107], with IDF values calculated from an atomic representation of the collection, same as for BM25F. This ensures we are only comparing the models in terms of satisfying the FI and TD constraints, rather than the TI-constraint, as otherwise the field specif IDF values

[1] https://www.kaggle.com/c/home-depot-product-search-relevance

would create noise. For BM25F, the version introduced in [69] where document length normalization is done at the document level, rather than the field level is used. This is because it represents a more concrete TFA model, where everything is done at the document level after summing the term frequencies together. Furthermore, analysing the effect of document length normalisation is not among the main objectives of this thesis. Field weights for each model are uniform.

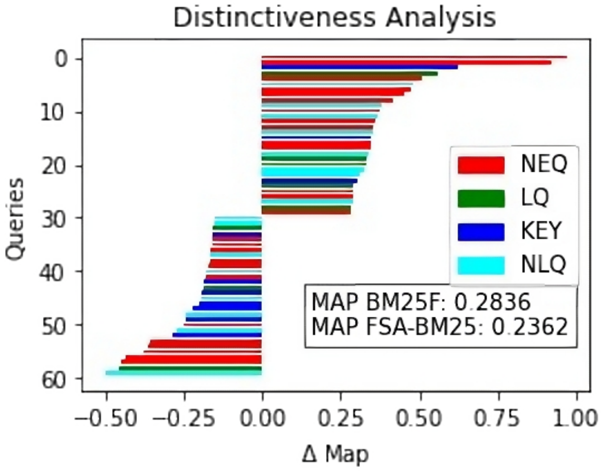


Figure 4.1: Comparison of BM25F and FSA-BM25 on DBpedia. $\Delta\,\text{MAP} = \text{MAP(BM25F)} - \text{MAP(FSA-BM25)}$.

Figure 4.1 compares the performance of BM25F and FSA-BM25. There are three important observations to be made: 1. purely in terms of MAP, BM25F performs much better than FSA-BM25 (0.284 vs. 0.236), 2. there are many queries where FSA-BM25 outperforms BM25F (up to 0.5 increase in MAP), meaning both models have their strengths and 3. there does not seem to be a trend for which of the query types do well for the models (colours are evenly distributed).

Possible reasons for why BM25F does better on some queries and FSA-BM25 on some were discussed in the previous chapter by looking at the top and bottom 5 queries in Table 3.2. To recap, BM25F does better for queries where the query terms would not necessarily occur in the same field and FSA-BM25 does better for queries where the query terms would occur all together in at least one field and where it is useful to consider term occurrences in different fields separately.

The above analysis confirms the importance of both the CO-TD and the CO-FD constraint. If an SDR model was able to saturate term frequencies across fields, i.e. to appreciate an occurrence of “steel” anywhere in the document more than a second occurrence of “concord”, whilst still considering term occurrences in different fields separately, we can clearly see from Figure 4.1, that there are significant performance gains to be made.

4.5.3 Field and Term Importance: Where do They Matter?

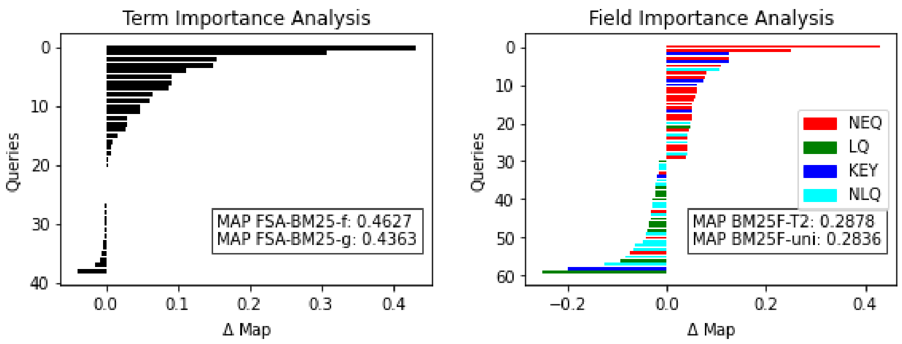


Figure 4.2: Comparison of model performance. LEFT (Homedepot):. 39 queries where $|q| = 1$, $\Delta\,$MAP $\neq 0$ and $\Delta\,$MAP = MAP(FSA-BM25-f) − MAP(FSA-BM25-g). RIGHT (DBpedia): $\Delta\,$MAP = MAP(BM25F-uni) − MAP(BM25F-T2).

For analysing the TI-constraints we compare two versions of FSA-BM25: One where the IDF component of the model is calculated from global document frequency values (**FSA-BM25-g**) and one where field-based document frequency values are used **(FSA-BM25-f**). Importantly, we only consider single-term queries. This is because the lack of term frequency saturation across fields results in more noise for the FSA-BM25-f model compared to the FSA-BM25-g model. This part of the analysis is only interested in the effect of the global v. fielded IDF. Therefore limiting the analysis to single-term queries does not affect the validity of the conclusions that are drawn. As DBpedia has a limited number of single-term queries we use the Homedepot dataset as discussed in Section 4.5.1 Figure 4.2 (left) compares the two models. From the figure it is evident that it is better to use field-based IDFs, thus confirming the importance of the CO-TI constraint.

For the field importance analysis, we compare the rankings of a BM25F model with uniform field weights (**BM25F-uni**) to one where the *title*-field has a weight of 2 (**BM25F-T2**) (other fields have a weight of 1). BM25F is used, rather than FSA-BM25 as it has been shown to perform better with field weighting [69]. Figure 4.2 (right) compares the performance of BM25F-uni and BM25F-T2 on DBpedia. There are 2 important observations to be made: 1.

The overall MAP is similar for both models (0.288 v. 0.284). 2. There is a trend in terms of the query types: BM25F-T2 performs well for the Named Entity Query type, which includes queries such as "Bradley Center" and "Plymouth Police Department" in the top 5, where the query terms appear in the *title*. BM25F-uni performs better for queries where the terms should not appear in the title, such as the list search queries: "did nicole kidman have any siblings?" and "matt berry tv series". In short, BM25-T2 works if we know that for a given query the *title*-field is important for the query in question, thus confirming the validity of the CO-FI constraint.

4.5.4 Discussion

The key points discussed in this section are 1. the trade-offs between the FD-Constraint and the TD-Constraint, 2. the relationship between IDF-CF-Rat$_{\mathrm{th}}(t, F_i, F_j, k_1)$ and IDF-CT-Rat$_{\mathrm{th}}(t, F_i, F_j, k_1)$, 3. query-type and domain considerations, and 4. what an SDR model that satisfies all four constraints would look like.

Table 4.3 illustrates the trade-off between satisfying the TD and FD-Constraints. Models that satisfy the FD-Constraint do not satisfy the TD-Constraint, and vice versa. As discussed by Robertson el al. FSA-based models assume independence of term frequencies across fields [21]. Regarding the example in Table 4.2, this means that it does not matter whether a document has both the query words "english" AND "spy", or just "english" spread over two fields, meaning the TD-constraint is not satisfied. TFA-based models solve this problem by saturating term frequency across fields. They assume a constant level of dependence between term occurrences in different fields, defined by their underlying scoring functions. However, in doing so they have to consider the document as atomic, rather than structured. In terms of the example in Table 4.2, this means that it does not matter whether a document has occurrences of "english" in the *plot* AND *description* fields, or just *description*, meaning the FD-Constraint is not satisfied.

Theorems 4.2 and 4.1 analyse the conditions for FSA-based models satisfying the FD-Constraint and TFA-based models satisfying the TD-Constraint. For FSA, the key metric to knowing whether a constraint is satisfied is the cross-field IDF ratio (Def. 4.7) and for TFA the cross-term IDF ratio (Def. 4.5). For each of these, there exists a threshold above which the FD-Constraint and the TD-Constraint are satisfied, respectively. If we simplify Eqns.(4.6) and (4.13) assuming the term frequencies from Table 4.2 and uniform field weights we get the simplified threshold values presented in Eqns. (4.19) and (4.11). Interestingly we observe that

$$\text{IDF-CF-Rat}_{\text{th}}(t, F_i, F_j, k_1) = \text{IDF-CT-Rat}_{\text{th}}(t_a, t_b, c, k_1) \tag{4.20}$$

meaning the cross-field IDF ratio threshold for satisfying the FD-Constraint for FSA is equal to the cross-term IDF ratio threshold for satisfying the TD-Constraint for TFA. For FSA the ratio is defined for a given term and for TFA between different terms. Whether each of the models satisfies their respective constraints depends on underlying collection statistics and the query. For example, if the query includes terms that have very different IDF values across fields, FSA models might not satisfy the FD-Constraint. Or, if the query terms have very different IDF values (some very rare and some common), TFA models might not satisfy the TD-Constraint. Which one is more likely, depends on the nature of the retrieval scenario. For example, in a QA retrieval scenario, it is likely that the query contains stopword-like terms. In such cases, not satisfying the TD-Constraint fully could be desirable. For keyword-like queries, the opposite is likely to be true. Not satisfying the FD-Constraint is more harmful in scenarios where there are many fields that carry different kinds of information, rather than in scenarios with redundant, or very similar fields.

The analysis in this chapter suggests that in order for an SDR model to satisfy the constraints (even conditionally), the model would need to facilitate term frequency saturation across fields (unlike the FSA), but should not revert to considering documents atomic (unlike the TFA). Furthermore, the model should consider the findings in Theorems 4.2 and 4.1, i.e. analytically asses the term specificity ratios at which the constraints are satisfied. This is exactly what is going to be done in the next chapter.

4.6 Summary, Conclusions and Contributions

This chapter covered the following issues:

- Introduction of four constraints for structured retrieval:
    - Term Distinctiveness: Adding unseen query terms to a document should increase the retrieval score more than adding query terms already considered.
    - Field Distinctiveness: Adding a query term to a new field should increase the retrieval score more than adding it to a field where it already occurs.
    - Term Importance: A model should consider the importance of a term on a field level, rather than a document level.

  - Field Importance: A model should be able to boost, or decrease the weight given to a field, based on some notion of field importance.

- In-depth analysis of when the existing models satisfy the constraints.
- Experimentation demonstrating why and how it is useful for models to satisfy the constraints.
- Discussion on how an SDR model could potentially satisfy all constraints.

The main conclusions were:

- The proposed four constraints lead to intuitive retrieval behaviour in simple retrieval scenarios.
- Satisfying the proposed four constraints is associated with better retrieval performance.
- For an SDR model to satisfy all four constraints, it needs to saturate term frequency across fields, whilst considering the field-level retrieval scores.

The main contributions are:

- The formalization of four retrieval constraints for SDR.
- Analysis demonstrating that they lead to intuitive ranking behaviour in simple retrieval scenarios.
- Analysis demonstrating if and how existing models satisfy the constraints, including formal theorems and proofs.
- Experimentation on established benchmarks, demonstrating that satisfying each constraint contributes to higher retrieval performance in its own way.

Chapter 5

Term Frequency Saturation in Information Content-based Field Weighting (ICFW)

This chapter introduces a version of information content-based field weighting, where term frequency is saturated across fields. Large parts of its content were published in ECIR'23 [3]. The chapter is structured as follows:

- Section 5.1 describes the motivation behind the chapter.
- Section 5.2 introduces the content and contributions of the chapter.
- Section 5.3 details how cross-field term frequency saturation is applied by the ICFW method.
- Section 5.4 explains how the lambda scaling parameter can be used to guarantee the satisfaction of SDR retrieval constraints introduced in the previous chapter.
- Section 5.5 describes how to best approximate the underlying collection metrics used to calculate lambda.
- Section 5.6 introduces a version of ICFW that can be better optimised.
- Section 5.7 presents the experimentation and analysis.
- Section 5.8 concludes.

5.1 Motivation

The high-level motivation for this chapter is the same as it was for Chapter 3; developing better analytical SDR models that can be used in investigative IR. This chapter can be seen as an iteration of Chapter 3 where the findings concerning the retrieval constraints for SDR are used to further develop the proposed approach, i.e. using information content for field weighting in SDR. One of the main findings of the previous chapter was that all four SDR constraints served a purpose and that satisfying them would require the saturation of term frequency across fields. This was the starting point for this section of the thesis: How can we develop the BM25-FIC model so that term frequency is not independent across fields? Furthermore, the goal was to make the weighting system that is not BM25 specific and instead have it work with any atomic retrieval model.

If it was possible to develop a model which could consider the structure of the documents explicitly (rather than just flattening the document as the BM25F does), whilst saturating term frequency across fields, this model would have the potential to outperform existing analytical SDR models significantly, as it would satisfy the SDR constraints. If such a model was developed and it was demonstrated that it outperforms existing models on a variety of data types, it would be a potential candidate for a new standard model in SDR.

5.2 Introduction

The main contribution of this chapter is a new field weighting method, denoted Information Content Field Weighting (ICFW). The method applies weights over the field-based scores produced by any atomic retrieval model (e.g. BM25, LM etc.) without optimization, just like the BM25-FIC. However, unlike BM25-FIC, ICFW saturates term frequency across fields. By setting the level of this saturation, it can be shown that ICFW satisfies all four SDR constraints from Chapter 4.

ICFW brings together all the lessons this thesis is looking to learn from atomic retrieval described in Section 2.8:

- **Exhaustivity:** The field-specific retrieval scores represent the exhaustivity of a field.
- **Specificity:** The specificity of a document field is estimated using the ICFW field weights.
- **Term frequency saturation:** The model saturates term frequency across fields.

- **Retrieval constraints:** The model is developed to satisfy all four SDR constraints.

5.3 Saturating Term Frequency Across Fields

ICFW aggregates the field-based retrieval scores of a document by multiplying each by their information content-based field weight and summing these weighted scores together. The field weight is calculated as a combination of collection field-based information content and document field-based information content, where a scale parameter λ determines the weight given to the latter. The more weight is given to document field-based information content, the more term frequency is saturated across fields.

Definition 5.1 (Term Probabilities)**.** *Let* $\mathrm{ff}(t, d)$ *be the field frequency; i.e. number of fields in* d *that contain term* t*.* $||F_i|| := \{f \big| |f| > 0\}$ *is the number of non-empty document fields. Let* $m(d) := \big|\{f | f \in d\}\big|$ *denote the number of fields in document* d*. The probability of a term occurring in a document field* f_i *(of type* i*) given collection field* F_i *is denoted* $P(t \in f_i|F_i)$*. The probability of a term occurring in a document field* f_i *given document* d *is denoted* $P(t \in f_i|d)$*.*

$$P(t \in f_i|F_i) \quad := \quad \frac{\mathrm{df}(t, F_i)}{||F_i||} \tag{5.1}$$

$$P(t \in f|d) \quad := \quad \frac{\mathrm{ff}(t, d)}{m(d)} \tag{5.2}$$

Note that Eqn. (5.1) *corresponds to Equation* (3.7)*, except empty fields are considered.*

Definition 5.2 (Field Probabilities)**.** *The probability of* q *and* f_i *given collection field* F_i *is denoted* $P(q, f_i|F_i)$*. The probability of* q *and* f_i *given document* d *is denoted* $P(q, f_i|d)$*.*

$$P(q, f_i|F_i) \quad := \quad \prod_{t \in q \cap f_i} P(t \in f_i|F_i) \tag{5.3}$$

$$P(q, f_i|d) \quad := \quad \prod_{t \in q \cap f_i} P(t \in f_i|d) \tag{5.4}$$

Definition 5.3 (ICF and ICD)**.** *The collection field-based information content of a document field f_i is denoted* $\mathrm{ICF}(q, f_i, F_i, d)$ *and the document-based information content of f_i is denoted* $\mathrm{ICD}(q, f_i, d)$*. Information content of an event is defined as its negative log probability as proposed by Hintikka [40] and used previously in the DFR model by Amati et al. [39].*

$$\begin{aligned} \mathrm{ICF}(q, f_i, F_i, d) &:= -\log P(q, f_i | F_i) & (5.5) \\ \mathrm{ICD}(q, f_i, d) &:= -\log P(q, f_i | d) & (5.6) \end{aligned}$$

If q is implicit and as F_i follows from f_i, $\mathrm{ICF}(q, f_i, F_i, d)$ *is shortened to* $\mathrm{ICF}(f_i, d)$ *and* $\mathrm{ICD}(q, f_i, d)$ *to* $\mathrm{ICD}(f_i, d)$*.*

Definition 5.4 (ICFW and Scale Parameter Lambda)**.** *Let λ be a scaling parameter defining the importance given to the document-based information content* ICD*. $\lambda \geq 0$.*

$$w_{\mathrm{icfw},\lambda_i}(f_i, F_i, d, q) := \mathrm{ICF}(q, f_i, F_i, d) + \lambda_i \cdot \mathrm{ICD}(q, f_i, d) \qquad (5.7)$$

where not ambiguous $w_{\mathrm{icfw}}(f_j, d)$ is short for $w_{\mathrm{icfw},\lambda_i}(f_i, F_i, d, q)$. Note that if $\lambda = 0$, ICFW is equal to BM25-FIC (apart from the $||F||$ variable).

Definition 5.5 (ICFW Retrieval Score)**.** *Let S_M be a retrieval score of retrieval model M (e.g. BM25). Given document d, query q, scaling parameter λ, collection c and retrieval model M, the score (retrieval status value) of d is denoted* $\mathrm{RSV}_{ICFW,\lambda,M}(d, q, c)$*.*

$$\mathrm{RSV}_{\mathrm{ICFW},\lambda,M}(d, q, c) := \sum_{i=1}^{m} w_{\mathrm{icfw},\lambda_i}(f_i, F_i, d, q) \cdot \mathrm{RSV}_M(q, f_i, c) \qquad (5.8)$$

The parameter λ scales the impact of the document-based information content. If λ is set to 0, $w_{\mathrm{icfw},\lambda_i}$ is defined only through information content based on the collection field (ICF), i.e. term occurrences would be considered independent between the fields as done in BM25-FIC in Chapter 3. As discussed in previous chapters and extensively by Robertson et al.[69], this is not a good assumption and results in the TD-constraint not being satisfied.

However, as λ increases, term frequency is saturated more across fields: Higher λ puts more emphasis on ICD, meaning it gives more weight to document fields with distinct terms, rather than ones re-appearing. I.e. the second occurrence of a term increases the retrieval score less than the first one, no matter what field it is in (assuming similar IDFs across fields). The size of λ defines the scale of this cross-field term frequency saturation.

The simplest way of setting lambda is to have it as a constant for the collection. In this way, the term frequency saturation across fields is constant, the same as for BM25F. Setting lambda this way will be considered in the experimentation. However, to find an appropriate value for lambda, optimization is needed. One of the main aims of this chapter was to provide a field weighting method that does not need optimization. The following section will describe an alternative way for setting λ that analytically considers the scale of term frequency saturation with respect to the TD-constraint and FD-constraint.

5.4 Satisfying SDR Constraints

The examples and analysis in this section largely assume the underlying field retrieval model for ICFW to be BM25. That is $M = \text{BM25}$ for $\text{RSV}_{\text{ICFW},M}$ (Definition 5.5). This assumption is made for the sake of clarity. The examples would work for any retrieval model (LM, DFR etc), but the math changes slightly between the models. There are two reasons for this: 1. For the TD-Constraint the ratio between term-level score contributions is important for the analysis. For the BM25, if we assume equal term frequencies for terms t_a and t_b, the ratio of their score contributions only depends on their IDF components. This makes the analysis more straightforward. 2. For the FD-Constraint we have a similar case where the ratio of term level score contributions between two levels of term frequency is important. For the BM25, this ratio depends only on the TF component, which again simplifies the analysis.

5.4.1 Satisfying the Term Distinctiveness Constraint

The aim of this section is to clarify how and when ICFW satisfies the TD-constraint. The following section does the same for the FD-constraint. Each section begins with a problem statement which frames the problem in a simpler manner with examples and visualizations. Afterwards, the problem is solved formally for the general case. The formal solution is then dissected more by simplifying some underlying assumptions and by observing how the model behaves.

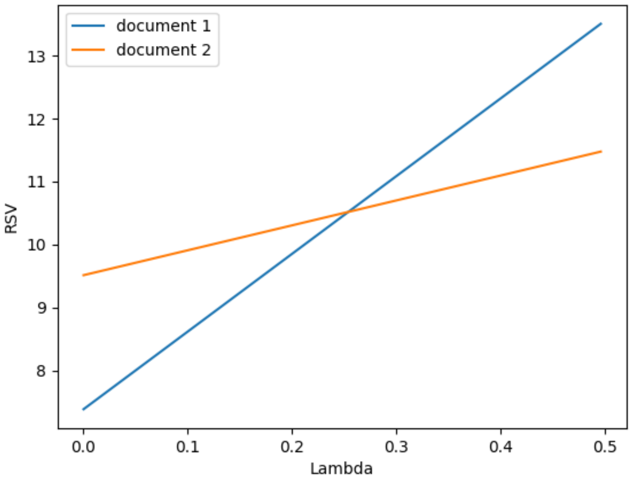
Problem Statement

As demonstrated in Chapter 4, TFA-based models (BM25F etc.) satisfy the TD-constraint because they saturate the term frequency across fields, however, in doing so they break the FD-constraint [2, 69, 70]. ICFW does not have this same problem, as term frequency can be saturated across fields, without reverting back to considering the document as atomic, as done by TFA-based models.

Lambda can be set for each query analytically, making sure the term frequency saturation is strong enough for the model to satisfy the TD-Constraint.

Figures 5.1 and 5.2 demonstrate how different values of λ affect the $\mathrm{RSV}_{\mathrm{ICFW}}$ scores with example documents. The linear regression-like nature of Equation (5.7) is evident from the figure and it also means the lines for documents 1 and 2 will cross. This is because the document with many occurrences of the rarer term (d2) will have a higher RSV score for $\lambda = 0$ and a less steep slope than a document with fewer occurrences of the rare term (d1). The slope for a document is defined by its document-based information content (ICD). Documents with fewer query terms, i.e. lower ICD will be ranked lower than documents with more query terms for high values of λ, meaning the TD-Constraint is satisfied.

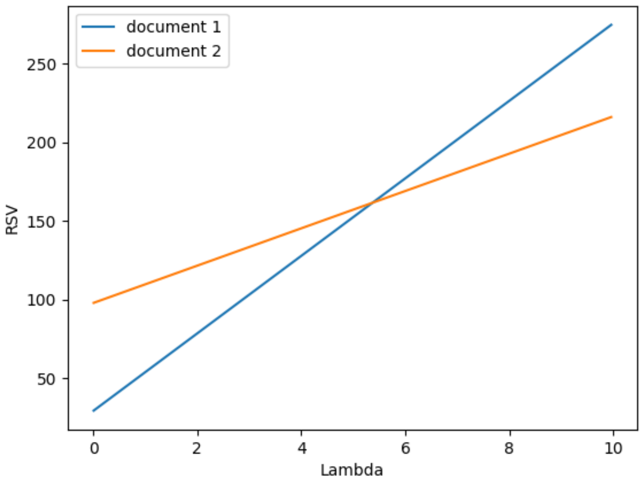
The first of the two figures (Figure 5.1) shows a scenario where the IDFs of query terms do not differ significantly. This means that the amount of term frequency saturation required is not very high and the lines cross at a relatively low level of lambda. Figure 5.2 shows a more severe case where the second occurrence of english in the description field is very rare. This means that cross-field term frequency saturation has to be high, meaning the lambda value where the lines cross is much higher for Figure 5.2 than for Figure 5.1.



field	plot		description		flattened doc
term	english	spy	english	spy	
field-specific IDF	1.9	2.0	2.5	2.1	
document 1	1	0	0	1	english spy
document 2	1	0	1	0	english english

Figure 5.1: Effect of lambda on RSV example. $m = 5$, $\lambda_{\text{TD-th}} = 0.28$

These examples suggest that there exists a threshold for lambda above which



field	plot		description		flattened doc
term	english	spy	english	spy	
field-specific IDF	3.8	3.7	9.1	3.9	
document 1	1	0	0	1	english spy
document 2	1	0	1	0	english english

Figure 5.2: Effect of lambda on RSV example. $m = 5$, $\lambda_{\text{TD-th}} = 5.9$.

the TD-constraint is satisfied. This threshold depends on query term and collection statistics. The next section will demonstrate how this threshold can be calculated for the general case, thus guaranteeing that ICFW satisfies the TD-constraint.

General Solution

In order to analyse and explain the conditionality of ICFW in satisfying the TD-constraint the following first generalises the question and answers it formally. The discussion that follows analyses the generalization in terms of issues that arise and simplifies some assumptions in order to communicate exactly how the TD-constraint is satisfied.

Definition 5.6 (Score Contribution of a Term)**.** *Let f denote a document field with occurrences of t and $\bar{f}$ denote an amended version of document field f without occurrences of t. The score contribution of a term t occurring in a field f is denoted as $\mathrm{S}_{\text{contr},M}(t, f, q, c)$. For clarity, where not ambiguous $\mathrm{S}_{\text{contr}}(t, f)$ is short for $\mathrm{S}_{\text{contr},M}(t, f, q, c)$.*

$$\mathrm{S}_{\text{contr},M}(t, f, q, c) := \mathrm{S}_M(q, f, c) - \mathrm{S}_M(q, \overline{f}, c) \tag{5.9}$$

Definition 5.7 (Score Contribution Ratios)**.** *Given terms* $\mathrm{t_a}$ *and* $\mathrm{t_b}$ *and document d, the cross-term score contribution ratio, i.e. the ratio of the score contributions (*$\mathrm{S_{contr}}$*) of terms* $\mathrm{t_a}$ *and* $\mathrm{t_b}$ *is denoted* $\Omega(\mathrm{t_a}, \mathrm{t_b}, f_i, f_j, d)$*. Given term t, fields* f_i *and* f_j*, the cross-field score contribution ratio, i.e. the ratio of the score contributions of t in fields* f_i *and field* f_j*, is denoted* $\Psi(t, f_i, f_j, d)$*.*

$$\Omega(\mathrm{t_a}, \mathrm{t_b}, f_i, f_j, d) := \frac{\mathrm{S_{contr}}(\mathrm{t_a}, f_i, d)}{\mathrm{S_{contr}}(\mathrm{t_b}, f_j, d)} \tag{5.10}$$

$$\Psi(t, f_i, f_j, d) := \frac{\mathrm{S_{contr}}(t, f_i, d)}{\mathrm{S_{contr}}(t, f_j, d)} \tag{5.11}$$

These definitions are the ICFW-equivalent of the Cross-Term IDF Ratio and Cross-Field IDF Ratio definitions from Chapter 4 where we considered the BM25F and BM25-FSA models and constraint satisfaction (Definition 4.5 and Definition 4.7).

Definition 5.8 (Scale TD-Threshold - Two Terms)**.** *Let* $q = \{t_1, \ldots, t_n\}$*, d a document with occurrences of term* t_a *in field* f_i *and occurrences of term* t_b *in field* $\overline{f}$*. Let* $\overline{d}$ *be an amended version of document d, where the occurrences of* t_b *in field* $\overline{f}$ *are replaced by further occurrences of term* t_a*. For presentation purposes* $\Omega(\mathrm{t_a}, \mathrm{t_b}, f_i, \overline{f}, d)$ *is shortened to* Ω *and* $\Psi(\mathrm{t_a}, \overline{f}, f_i, d)$ *to* Ψ*. When used inside equations, the terms are sometimes further simplified to* Ω *and* Ψ*.*

$$\lambda_{\mathit{TD\text{-}th}}(\mathrm{t_a}, \mathrm{t_b}, d, f_i) := \frac{\log \frac{\mathrm{df}(\mathrm{t_b}, \overline{F}) |\overline{F}|^{\Omega\Psi}}{\mathrm{df}(\mathrm{t_a}, \overline{F})^{\Omega\Psi} |\overline{F}|}}{\log \frac{m^{\Omega+1}\, \mathrm{ff}(\mathrm{t_a}, \overline{d})^{\Omega(\Psi+1)}}{m^{\Omega(\Psi+1)}\, \mathrm{ff}(\mathrm{t_a}, d)^{\Omega+1}}} \tag{5.12}$$

$$\left(= \frac{\Omega\Psi\, \mathrm{ICF}(\overline{f}, \overline{d}) - \mathrm{ICF}(\overline{f}, d)}{-\Omega(\Psi+1)\, \mathrm{ICD}(f_i, \overline{d}) + \Omega\, \mathrm{ICD}(f_i, d) + \mathrm{ICD}(\overline{f}, d)} \right) \tag{5.13}$$

$\lambda_{\text{TD-th}}$ defines the lambda-value above which $\mathrm{score}(d) > \mathrm{score}(\overline{d})$. This is the lambda-value above which the ICFW satisfies the TD-constraint with respect to t_a and t_b. See formal theorem and proof in Appendix A.1. The following generalizes Definition 5.8 for the entire query, rather than two specific terms.

Definition 5.9 (Scale TD-Threshold - Query)**.** *In order to generalize Def. 5.8 to the entire query, rather than two query terms and* f_i*, we need to consider the rarest and most common query terms and the field with the smallest* $\mathrm{S_{contr}}(\mathrm{t_a}, f_i, d)$*. Let* $\mathrm{t_{ra}}$ *be the rarest query term,* $\mathrm{t_{co}}$ *the most common query term and* $f_{\min}$ *the field with the smallest score contribution (*$\mathrm{S_{contr}}(\mathrm{t_{ra}}, f_i, d)$*) for term* $\mathrm{t_{ra}}$*.*

$$\lambda_{\mathit{TD\text{-}th}}(q, d) := \lambda_{\mathit{TD\text{-}th}}(\mathrm{t_{ra}}, \mathrm{t_{co}}, d, f_{\min}) \tag{5.14}$$

Setting $t_a = \mathrm{t_{ra}}$, $t_b = \mathrm{t_{co}}$, $f_i = f_{\min}$ ensures $\mathrm{score}(d) > \mathrm{score}(\overline{d})$ for the entire document and query. This is because changing the most common term to the rarest term in $\overline{f}$ has the highest impact on $\mathrm{RSV}(q, d, c)$, which needs to be offset by the ICD component and therefore a larger value for λ.
From the Theorem we can see that Definition 5.8 holds if:

$$\frac{\mathrm{ICF}(\overline{f}, d)}{\Psi\,\mathrm{ICF}(\overline{f}, \overline{d})} < \Omega < \frac{\mathrm{ICD}(\overline{f}, d)}{(\Psi + 1)\,\mathrm{ICD}(f_i, \overline{d}) - \mathrm{ICD}(f_i, d)} \tag{5.15}$$

$$\frac{\log \frac{\mathrm{df}(t_b, \overline{F})}{|\overline{F}|}}{\Psi \log \frac{\mathrm{df}(t_a)}{|\overline{F}|}} < \Omega < \frac{-\log \frac{\mathrm{ff}(t_b, d)}{m}}{\log \frac{\mathrm{ff}(t_a, d)}{m} - (\Psi + 1) \log \frac{\mathrm{ff}(t_a, \overline{d})}{m}} \tag{5.16}$$

The discussion section that follows will examine this conditionality in more depth.

Discussion

As mentioned at the beginning of this section Definition 5.8 is far from intuitive, or straightforward. It is worth dissecting it more by making some simplifying assumptions. By setting $\Psi(\mathrm{t_a}, \overline{f}, f_i, d) = 1$, meaning we assume that terms have equal score contributions across fields, the formula simplifies to

$$\lambda_{\text{TD-th}}(\mathrm{t_a}, \mathrm{t_b}, d, f_i) = \frac{\log \frac{\mathrm{df}(\mathrm{t_b}, \overline{F})|\overline{F}|^{\Omega}}{\mathrm{df}(\mathrm{t_a}, \overline{F})^{\Omega}|\overline{F}|}}{\log \frac{m^{\Omega+1}\,\mathrm{ff}(\mathrm{t_a}, \overline{d})^{2\Omega}}{m^{2\Omega}\,\mathrm{ff}(\mathrm{t_a}, d)^{\Omega+1}}} \tag{5.17}$$

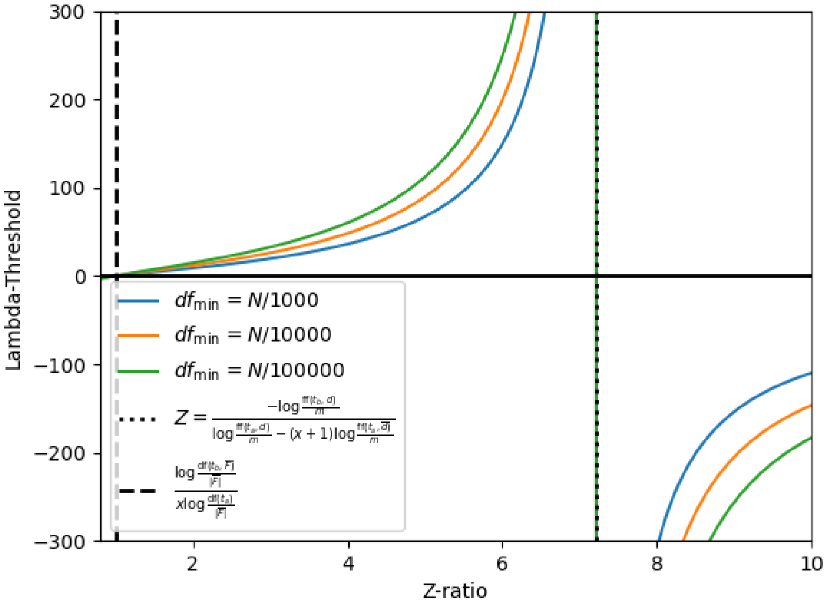
If we further assumes that $\Omega(\mathrm{t_a}, \mathrm{t_b}, f_i, \overline{f}, d) = 1$ Equation 5.17 simplifies to

$$\lambda_{\text{TD-th}}(\mathrm{t_a}, \mathrm{t_b}, d, f_i) = \frac{\log \frac{\mathrm{df}(\mathrm{t_b}, \overline{F})|\overline{F}|}{\mathrm{df}(\mathrm{t_a}, \overline{F})|\overline{F}|}}{\log \frac{m^2\,\mathrm{ff}(\mathrm{t_a}, \overline{d})^2}{m^2\,\mathrm{ff}(\mathrm{t_a}, d)^2}} = 0 \tag{5.18}$$

By setting $\Psi = 1$ and $\Omega = 1$ we assume that all terms have the same IDF in all fields. This is not a realistic assumption, but it does resemble the assumptions made by Fang et al. with respect to their formal retrieval constraints: Their TFC3-Constraint states that documents with more distinct query terms should be favoured, given equal IDFs (specificities). In our context, this is the same as assuming $\Omega = 1$ and $\Psi = 1$

Equation (5.18) shows that if there are no differences in the score contributions of terms across terms or fields, as long as λ is set above 0 the TD-Constraint is satisfied. As we constraint lambda to $\lambda > 0$ in Definition 5.4 it can be said that if $\Omega = 1$ and $\Psi = 1$, ICFW always satisfies the TD-Constraint.

Figure 5.6 visualizes the lambda threshold changes for different values of Ω. The curve is a rectangular hyperbola where the Ω value at the vertical asymptote and at $\lambda_{\text{TD-th}} = 0$ are of special interest to us in terms of the lambda threshold and the TD-Constraint. We assume that $T = z$ with respect to Theorem 4.2. This is something that is inherently assumed by the SDR constraints and is therefore a simplification that could have been made sooner. However, it was determined that for clarity it was more sensible in the previous section to solve the problem for a more general case and simplify it later. Consider first the



field	plot		description		flattened doc
term	english	spy	english	spy	
field-specific IDF	2.0	2.0	2.0	2.0	
document 1	1	0	0	1	spy english
document 2	1	0	1	0	english english

Figure 5.3: Effect of lambda on RSV example. $x = 1$, $T = z$, $m = 5$, $\frac{-\log\frac{\text{ff}(t_b,d)}{m}}{\log\frac{\text{ff}(t_a,d)}{m} - (\Psi+1)\log\frac{\text{ff}(t_a,\bar{d})}{m}} = 7.212$

and $\frac{\text{ICF}(\bar{f},d)}{\Psi\,\text{ICF}(\bar{f},\bar{d})} = 1$

point where $\lambda_{\text{TD-th}} = 0$. If we assume that $x = 1$, it will always be the case that $\frac{\text{ICF}(\bar{f},d)}{x\,\text{ICF}(\bar{f},\bar{d})} = 1$. This is because the Ω-ratio will be defined by the IDF values alone and at $\Omega = 1$ the IDFs being equal means the DFs are equal as well. Put in another way, as Ω is defined for a rare and a common term, if $\Omega < 1$, the rare term becomes the common term and the common term becomes the rare term.

The asymptote point at $\frac{-\log\frac{\text{ff}(t_b,d)}{m}}{\log\frac{\text{ff}(t_a,d)}{m} - (\Psi+1)\log\frac{\text{ff}(t_a,\bar{d})}{m}}$ defines a limit to where $\lambda_{\text{TD-th}}$ can be defined so that the TD-Constraint is satisfied. Meaning the condition for ICFW satisfying the TD-Constraint is that there exists a lambda-TD-threshold such that $\lambda_{\text{TD-th}} >= 0$ and $\lambda_{\text{TD-th}} < \frac{-\log \frac{\text{ff}(t_b,d)}{m}}{\log \frac{\text{ff}(t_a,d)}{m} - (\Psi+1) \log \frac{\text{ff}(t_a,\bar{d})}{m}}$. It is worth mentioning that the threshold for the second condition is highly dependent on m and for high values of m the value is notably smaller. To summarize, within reasonable assumptions about the underlying model (satisfies TC1-constraint by Fang et al.) and assuming $\Psi = 1$, ICFW satisfies the TD-Constraint as long as $\lambda > \lambda_{\text{TD-th}}$.

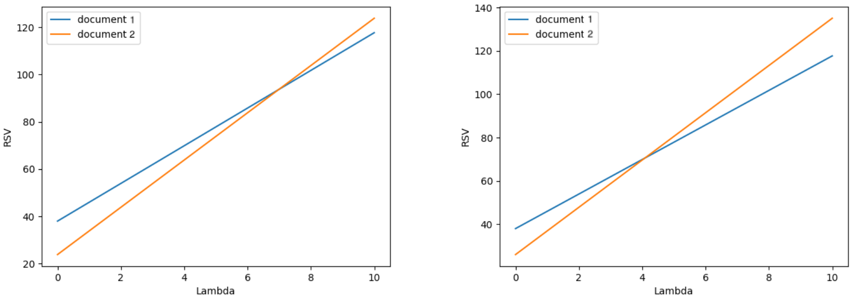
5.4.2 Satisfying the Field Distinctiveness Constraint

This section analyses how ICFW satisfies the FD-Constraint much in the same way as the previous section did for the TD-Constraint, starting with a problem statement, solving the problem for the general case and then clarifying the solution by simplifying some underlying assumptions using examples.

Problem Statement

As discussed in Chapter 4, FSA-based models satisfy the FD-Constraint conditionally. Theorem 4.2 demonstrated how FSA-BM25 failed to satisfy the FD-Constraint if the cross-field IDF ratio was below a certain level. That is if the IDF of a term is much higher in one field, it is possible that the FSA-BM25 score of a document where a term occurs in only one field is higher than the score of a document where that term occurs in many fields. A similar conditionality exists for ICFW.

Figures 5.4 and 5.5 demonstrate lambda affects the RSV_{ICFW} for example documents. They are similar to the figures seen in the previous section. Figure 5.4 shows an example where the IDF values for the terms are relatively similar. It is worth pointing out that the term “spy” is in fact irrelevant here since the FD-Constraint is only concerned with singular terms. The term is kept in the table for consistency with the previous section. An important aspect of the analysis here is the fact that there are term frequencies greater than one in the table in Figure 5.4. This means that the within-field term frequency saturation plays an important part. Therefore the figures in this section consider the example scenario for different values of k_1. As discussed in Chapter 4, the desired ranking of the documents in Figure 5.4 is $\text{score}(d1) > \text{score}(d2)$. From the figure, it is clear that this is indeed the case as long as lambda is below a certain threshold (where the lines cross). It is also clear that the degree of the within-field term frequency saturation (k_1) has a significant effect on this threshold. If lambda is too high, the score of document 1 falls below that of document 2.



$k_1 = 1.2$ $k_1 = 2.0$

field	plot		description		flattened doc
term	english	spy	english	spy	
field-specific IDF	4.2	4.3	4.5	4.2	
document 1	1	0	1	0	english english
document 2	0	0	2	0	english english

Figure 5.4: Effect of lambda on RSV example. $m = 5$

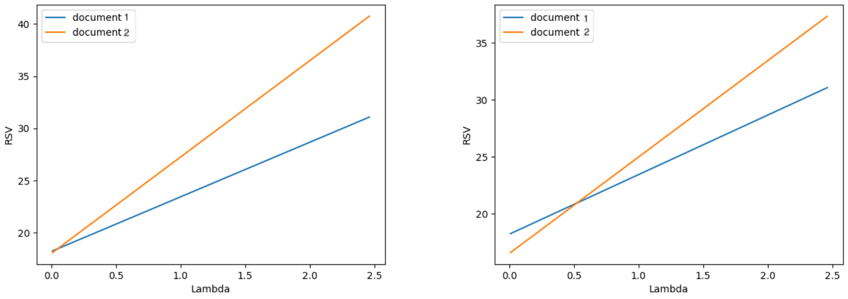
Figure 5.5 shows a similar analysis for a different scenario, where the IDF value for “english” is much higher in the “description” field. It is easy to see that the lambda for where the documents are ranked correctly is significantly lower.

These examples suggest that there exists a threshold for lambda below which the FD-constraint is satisfied. This threshold depends on query term and collection statistics, as well as the degree of within-field term frequency saturation. The next section will demonstrate how this threshold can be calculated for the general case, thus guaranteeing that ICFW satisfies the FD-Constraint.

General Solution

In order to analyse and explain the conditionality of ICFW in satisfying the FD-constraint the following first generalises the question and answers it formally. The discussion that follows analyses the generalization in terms of issues that arise and simplifies some assumptions in order to communicate exactly how the FD-constraint is satisfied.

Definition 5.10 (Score Contribution Ratio)**.** *Let* $q = \{t_1, \ldots, t_n\}$ *be a query,* d *a document with* T *occurrences of term* t *in field* f_i *and* z *occurrences of term* t *in another field* f_j*. Let* $\overline{d}$ *be an amended version of* d*, where the occurrences of term* t *in* f_j *have been moved to* f_i *and* z *occurrences of non-query terms have removed from* f_i *and added to* f_j*. These non-query terms ensure that the theorem is only concerned with query term occurrences, rather than document*



$k_1 = 1.2$ $k_1 = 2.0$

field	plot		description		flattened doc
term	english	spy	english	spy	
field-specific IDF	1.9	2.0	4.0	1.2	
document 1	1	0	1	0	english english
document 2	0	0	2	0	english english

Figure 5.5: Effect of lambda on RSV example. $m = 5$

lengths. For clarity $\zeta(t, f_i, d)$ is sometimes simplified to just ζ.

$$\zeta(t, f_i, d) := \frac{\mathrm{S}_{\mathrm{contr}}(t, f_i, \overline{d})}{\mathrm{S}_{\mathrm{contr}}(t, f_i, d)} \tag{5.19}$$

Definition 5.11 (FD-Constraint Scale Threshold - Two Fields)**.** *Let $q = \{t_1, \ldots, t_n\}$ be a query, d a document with T occurrences of term t in field f_i and z occurrences of term t in an average field $\overline{f}$. Let $\overline{d}$ be an amended version of d, where the occurrences of term t in $\overline{f}$ have been moved to f_i and z occurrences of non-query terms have removed from f_i and added to $\overline{f}$. These non-query terms ensure that the theorem is only concerned with query term occurrences, rather than document lengths.*

$$\lambda_{FD\text{-}th}(t, d, f_i, \overline{f}) :=$$

$$\frac{(\zeta - 1)\,\mathrm{ICF}(f_i, d) - \Psi\,\mathrm{ICF}(\overline{f}, d)}{(1 + \Psi)\,\mathrm{ICD}(f_i, d) - \zeta\,\mathrm{ICD}(f_i, \overline{d})} \quad \left(= \frac{\log \frac{[\frac{\mathrm{df}(t,\overline{F})}{|\overline{F}|}]^{\Psi - 1}}{[\frac{\mathrm{df}(t,F_i)}{|F_i|}]^{\zeta - 1}}}{\log \frac{[\frac{\mathrm{ff}(t,\overline{d})}{m}]^{\zeta}}{[\frac{\mathrm{ff}(t,d)}{m}]^{1+\Psi}}} \right) \tag{5.20}$$

$\lambda_{\text{FD-th}}$ defines the λ value above which $\mathrm{score}(d) > \mathrm{score}(\overline{d})$. This is the λ above which the ICFW satisfies the FD-Constraint with respect to t, f_i and $\overline{f}$. See Appendix A.2 for formal theorem and proof. From the Theorem we can say that Definition 5.11 holds if

$$(1+\Psi)\frac{\log\frac{\mathrm{ff}(t,d)}{m}}{\log\frac{\mathrm{ff}(t,\overline{d})}{m}} < \zeta < \frac{\Psi\log\frac{\mathrm{df}(t,\overline{F})}{N}}{\log\frac{\mathrm{df}(t,F_i)}{N}} + 1 \tag{5.21}$$

$$\frac{(1+\Psi)\,\mathrm{ICD}(f_i,d)}{\mathrm{ICD}(f_i,\overline{d})} < \zeta < \frac{\Psi\,\mathrm{ICF}(\overline{f},d)}{\mathrm{ICF}(f_i,d)} + 1 \tag{5.22}$$

The following generalizes Definition 5.11 to the entire query and collection rather than a single term and two fields.

Definition 5.12 (FD-Constraint Scale Threshold - Query)**.** *Let the t_{highpsi} be a term with the highest Ψ-ratio and let f_{com} and f_{rare} be the two fields between which the highest Ψ-ratio is defined.*

$$\lambda_{\mathit{FD\text{-}th}}(q,d) := \lambda_{\mathit{FD\text{-}th}}(t_{\text{highx}}, d, f_{\text{com}}, f_{\text{rare}}) \tag{5.23}$$

The next section, together with Appendix A.2 will clarify this conditionality using examples.

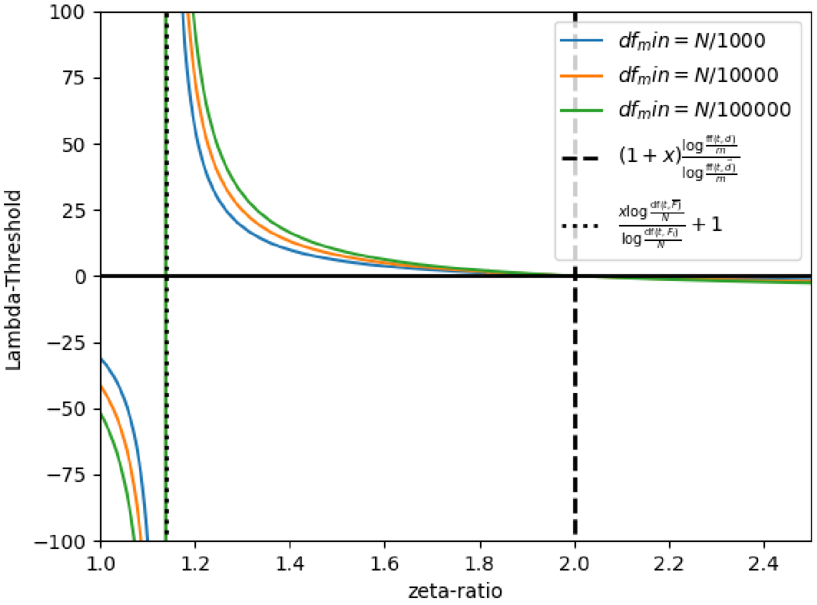
Discussion

As mentioned at the beginning of this section Definition 5.11 is far from intuitive, or straightforward. It is worth dissecting it more by making some simplifying assumptions. Consider first the score contribution ration ζ (Equation 5.19). Assuming BM25 as the underlying retrieval model and equal document lengths, ζ is only dependent on the term frequency and the parameter k_1 as the IDF components cancel out:

$$\zeta(t,f_i,d) = \frac{\mathrm{TF}_{\mathrm{BM25}}(t,d,c|n(t,d)=T+z)}{\mathrm{TF}_{\mathrm{BM25}}(t,d,c|n(t,d)=T)} \tag{5.24}$$

Figure 5.6 visualizes the lambda threshold changes for different values of ζ. The curve is a rectangular hyperbola where ζ value at the vertical asymptote and at $y = 0$ are of special interest to us in terms of the lambda threshold and the FD-Constraint. This is because $\lambda > 0$ only between these two points on the curve. As this was one of the initial assumptions for the ICFW model, it can be said that ICFW can only satisfy the FD-Constraint if there exists a lambda-threshold value between these two points.

Consider first the dotted line on the left, i.e. the vertical asymptote. If ζ is lower than this value, the inequality in Equation (5.11) is reversed (See Theorem A.2). This means that in order for the FD-Constraint to be satisfied we must have $\lambda > \lambda_{\text{FD-th}}$. This is also the point where $\lambda_{\text{FD-th}}$ becomes negative, meaning our base assumption of $\lambda > 0$ from Definition 5.4 is enough to guarantee



field	plot		description		flattened doc
term	english	spy	english	spy	
field-specific IDF	2.0	2.0	2.0	2.0	
document 1	1	0	1	0	english english
document 2	0	0	2	0	english english

Figure 5.6: Equation 5.20 plotted against ζ. Assume $\Psi = 1$, $T = z$ and $m = 5$.

that lambda is greater than the value of the asymptote.

The dashed line in Figure 5.6 is another point where $\lambda_{\text{FD-th}}$ becomes negative, meaning it is not defined for our model. In order to better understand the meaning of this consider the example documents in Figure 5.6. $\zeta > 2$ would mean that the two occurrences of "english" for document 2 would have more than twice the score contribution of one occurrence of "english" in document 1. This means that not only would there be no within-field term frequency saturation, but the second occurrence of a term would be actually given more weight than the second occurrence one. This will not happen if we assume that the underlying retrieval model is sensible (i.e. some term frequency saturation) and we assume that $T = z$ in Theorem 4.1. As discussed earlier, the latter is an assumption made in the definition of the SDR constraints and therefore valid. It is worth mentioning that the above analysis assumes that $\Psi = 1$. This is not a valid assumption, however, it does help clarify how the model works.

5.4.3 Satisfying the TD-Constraint and the FD-Constraints Together

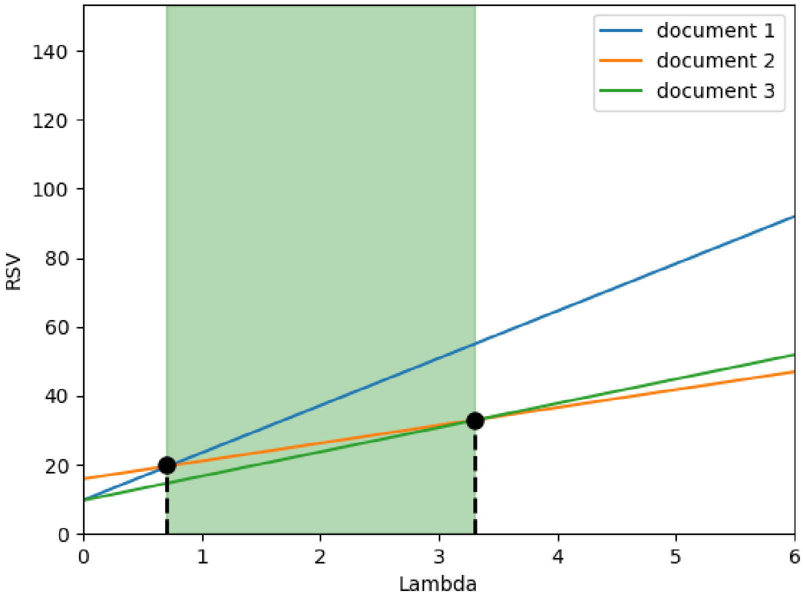
The preceding two sections have shown that ICFW satisfies the TD and FD-Constraints given some conditions. The purpose of this section is to examine whether the satisfaction of one constraint affects the satisfaction of the other. This is an important point, as Chapter 4 showed how there is an inherent trade-off between the two aggregation functions for existing SDR models. Furthermore, both constraints were conditional on the value of λ, suggesting there might be trade-offs.

For the sake of clarity, it is worth starting with the more straightforward case where we assume that query terms have equal specificities (IDFS) across fields. As mentioned earlier, this is not a valid assumption, but it has been used in the past in retrieval constraint research and it provides a good starting point. Assuming equal IDFs across fields is the same as assuming that $\Psi = 1$ and $\Omega = 1$ within the context of lambda-threshold definitions (Definitions 5.8 and 5.11) and the constraint definitions in Chapter 4 (4.1 and 4.2). Equation (5.18) shows that with these assumptions as long $\lambda > 0$, the constraint is satisfied. Since lambda is defined as $\lambda > 0$ in Definition 5.4, ICFW satisfies the TD-constraint unconditionally given the above assumptions. From the analysis in the previous section, we can say that given the above assumptions, ICFW also satisfies the FD Constraint as long as $\zeta < \frac{\log \frac{\mathrm{df}(t,\overline{F})}{N}}{\log \frac{\mathrm{df}(t,F_i)}{N}} + 1$ or $\lambda < \lambda_{\text{FD-th}}$. In short, assuming equal IDFs across fields, both the TD-Constraint and FD-Constraint are satisfied if $\lambda > 0$ and $\lambda < \lambda_{\text{FD-th}}$.

In a real retrieval scenario we would expect the IDF values to vary across terms, as well as fields. The examples in Figures 5.7 and 5.8 will examine how the FD and TD-Constraints interact in such scenarios. Figure 5.7 shows an example where there are differences in the IDF values, but they are still relatively small. We can see that there exist lambda values where $\lambda_{\text{FD-th}} < \lambda < \lambda_{\text{TD-th}}$, which means that if we set lambda as such, both the TD and FD-Constraints are satisfied.

5.4.4 Satisfying the Term Importance Constraint

ICFW satisfies the TI-Constraint as it considers the specificity of terms separately depending on which field they occur in. In the example above this means that if $\mathrm{IDF}(\mathrm{english}, \mathrm{description}) > \mathrm{IDF}(\mathrm{english}, \mathrm{plot})$, more importance is given to the occurrence of "english" in the description field. This is the case for two reasons: 1. the collection-based information content (ICF) is higher for documents where "english" occurs in the description field and 2. because the



field	plot		description		flattened doc
term	english	spy	english	spy	
field-specific IDF	2.7	2.0	2.9	1.5	
document 1	1	0	0	1	english spy
document 2	1	0	1	0	english english
document 3	0	0	2	0	english english

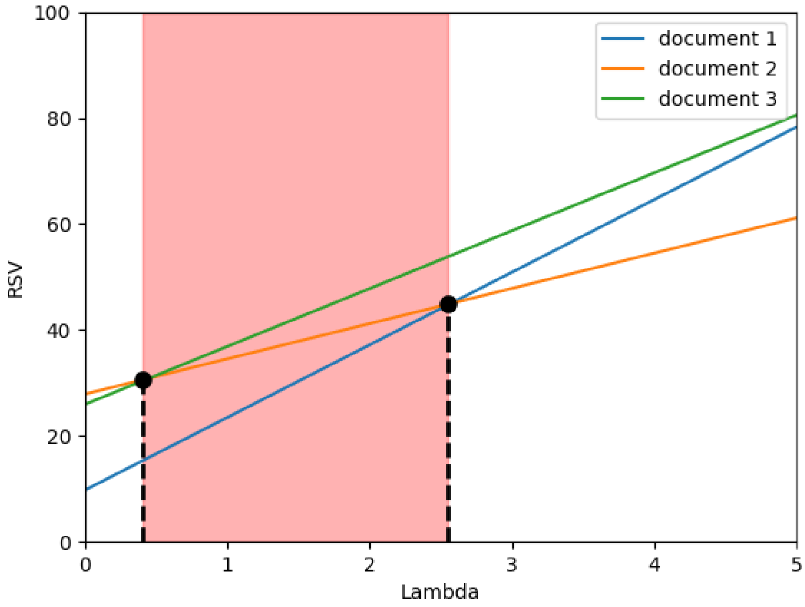
Figure 5.7: Effect of lambda on ranking example. $m = 5$, $k_1 = 2.0$

underlying retrieval model (BM25, LM, DFR) should also give more weight to term occurrences with higher specificity (assuming the retrieval model chosen is “sensible”).

5.4.5 Satisfying the Field Importance Constraint

ICFW satisfies the FI-Constraint as information content is seen as a measure of the importance of a field. Documents with term occurrences in fields with higher information content a favoured, and thus the constraint is satisfied.

Figure 5.8 shows a scenario where the occurrence of “english” has a significantly higher IDF than “spy” in the description field. This results in a situation where there exists no lambda such that $\lambda_{\text{FD-th}} < \lambda < \lambda_{\text{TD-th}}$, meaning only one of the constraints can be satisfied.



field	plot		description		flattened doc
term	english	spy	english	spy	
field-specific IDF	2.7	2.0	4.5	1.5	
document 1	1	0	0	1	english spy
document 2	1	0	1	0	english english
document 3	0	0	2	0	english english

Figure 5.8: Effect of lambda on ranking example. $m = 5$, $k_1 = 2.0$

5.4.6 Constraint Satisfaction Summary

Table 5.1 summarizes the satisfaction of the SDR constraints by ICFW and the existing models discussed in Chapter 4.

The conditionality for ICFW satisfying the TD-Constraint and FD-Constraint respectively are as follows:

Definition 5.13 (TD-Constraint Conditionality)**.** *ICFW satisfies the TD-Constraint if:*

$$0 < \lambda < \lambda_{FD\text{-}th} \ AND \ \Omega < \frac{-\log \frac{\mathrm{ff}(t_b,d)}{m}}{\log \frac{\mathrm{ff}(t_a,d)}{m} - (x+1)\log \frac{\mathrm{ff}(t_a,\bar{d})}{m}} \tag{5.25}$$

Definition 5.14 (FD-Constraint Conditionality)**.** *ICFW satisfies the FD-Constraint if the field-level retrieval model saturates term frequency and if:*

$$0 < \lambda < \lambda_{FD\text{-}th} \tag{5.26}$$

	Term Distinct. TD-Co	Field Distinct FD-Co	Term Import. TI-Co	Field Import. FI-Co
FSA	NO	Conditional	YES	YES
BM25-FIC	NO	Conditional	YES	YES
PRMS	NO	Conditional	NO	YES
BM25F	Conditional	NO	NO	YES
MLM	Conditional	NO	NO	YES
FSDM	Conditional	NO	NO	YES
ICFW	**Conditional**	**Conditional**	**YES**	**YES**

Table 5.1: Constraint satisfaction of SDR models, including ICFW.

If both the conditions above are satisfied, then both TD and FD-Constraints are satisfied. As we can see, ICFW is the only one that satisfies all four constraints (given the conditions described above). Analytically speaking this is the reason we would expect it to perform better than the other models.

5.5 Approximating Appropriate Values for TD-Constraint Threshold

If used directly Definition 5.9 is highly sensitive to per-term score contributions (Definition 5.6). A single query term that is very rare in one of the fields defines λ for all documents. This is because Definition 5.9 sets the lambda-threshold so that the second occurrence of such a rare term needs to be offset by a first occurrence of a common term. Therefore, Definitions 5.9 should not be viewed as a definition of an optimal lambda value, but as a good starting point with intuitive explanations.

With this in mind, the calculation of the thresholds is made less sensitive to large variations of term specificity in Definitions 5.10 and 5.11. Firstly, we assume that $\Psi = 1$, i.e. the specificity of a term is assumed to be the same in all fields. This is done as we do not want λ to become too sensitive to variations in specificity for a single term across fields. Rather we are more interested in satisfying the TD-constraint and thus care more about the specificity values across terms. Second, the effect of metrics based on singular terms and fields needs to be smoothed using the rest of the query terms and the collection. The three proposed methods for this smoothing approximate the df values in Definition 5.8 in terms of the document frequencies and in terms of the $\mathrm{S}_{\mathrm{contr}}$ values in Definition 5.10 resulting in three proposed models:

Definition 5.15 (ICFW-Global (ICFW-G)). *Let* $n(\mathrm{t}_{\mathrm{ra}}, f) = 1$ *and*

$n(\mathrm{t}_{\mathrm{co}}, f) = 1$.

$$\mathrm{df}_{\text{ICFW-G}}(\mathrm{t}_{\mathrm{ra}}, F_i) = \min(\{\mathrm{df}(t, c) : t \in q\}) \tag{5.27}$$
$$\mathrm{df}_{\text{ICFW-G}}(\mathrm{t}_{\mathrm{co}}, F_i) = \max(\{\mathrm{df}(t, c) : t \in q\}) \tag{5.28}$$
$$\mathrm{S}_{\text{contr,ICFW-G}}(\mathrm{t}_{\mathrm{co}}, f_i, d) = \min(\{\mathrm{S}_{\text{contr,G}}(t, f_i, d) : t \in q\}) \tag{5.29}$$
$$\mathrm{S}_{\text{contr,ICFW-G}}(\mathrm{t}_{\mathrm{ra}}, f_i, d) = \max(\{\mathrm{S}_{\text{contr,G}}(t, f_i, d) : t \in q\}) \tag{5.30}$$

Definition 5.16 (ICFW-Global-Average (ICFW-GA)). *Let $t_{\max}$ be the most common query term in the collection, $n(\mathrm{t}_{\mathrm{ra}}, f) = 1$ and $n(\mathrm{t}_{\mathrm{co}}, f) = 1$.*

$$\mathrm{df}_{\text{ICFW-GA}}(\mathrm{t}_{\mathrm{ra}}, F_i) = \frac{\sum_{t \in q \setminus t_{\max}} \mathrm{df}(t, c)}{|t \in q \setminus t_{\max}|} \tag{5.31}$$
$$\mathrm{df}_{\text{ICFW-GA}}(\mathrm{t}_{\mathrm{co}}, F_i) = \max(\{\mathrm{df}(t, c) : t \in q\}) \tag{5.32}$$
$$\mathrm{S}_{\text{contr,ICFW-GA}}(\mathrm{t}_{\mathrm{co}}, f_i, d) = \frac{\sum_{t \in q \setminus t_{\max}} \mathrm{S}_{\text{contr,GA}}(t, f_i, d)}{|t \in q \setminus t_{\max}|} \tag{5.33}$$
$$\mathrm{S}_{\text{contr,ICFW-GA}}(\mathrm{t}_{\mathrm{ra}}, f_i, d) = \max(\{\mathrm{S}_{\text{contr,GA}}(t, f_i, d) : t \in q\}) \tag{5.34}$$
$$\tag{5.35}$$

Definition 5.17 (ICFW-Local-Average (ICFW-LA)). *Let $n(\mathrm{t}_{\mathrm{ra}}, f) = 1$ and $n(\mathrm{t}_{\mathrm{co}}, f) = 1$.*

$$\mathrm{df}_{\text{ICFW-LA}}(\mathrm{t}_{\mathrm{ra}}, F_i) = \frac{\sum_{t \in q \setminus t_{\max}} \mathrm{df}(t, F_i)}{|t \in q \setminus t_{\max}|} \tag{5.36}$$
$$\mathrm{df}_{\text{ICFW-LA}}(\mathrm{t}_{\mathrm{ra}}, F_i) = \max(\{\mathrm{df}(t, c) : t \in q\}) \tag{5.37}$$
$$\mathrm{S}_{\text{contr,ICFW-LA}}(\mathrm{t}_{\mathrm{co}}, f_i, d) = \frac{\sum_{t \in q \setminus t_{\max}} \mathrm{S}_{\text{contr,LA}}(t, f_i, d)}{|t \in q \setminus t_{\max}|} \tag{5.38}$$
$$\mathrm{S}_{\text{contr,ICFW-LA}}(\mathrm{t}_{\mathrm{ra}}, f_i, d) = \max(\{\mathrm{S}_{\text{contr,LA}}(t, f_i, d) : t \in q\}) \tag{5.39}$$

ICFW-G uses the document frequency values over the whole collection. ICFW-GA further smooths the effect of rare query terms by estimating $\mathrm{df}(\mathrm{t}_{\mathrm{ra}}, c)$ as the mean of collection-level document frequencies of terms that are not the most common. ICFW-LA is similar to ICFW-GA, except the calculations are done at the collection field level, rather than the collection level (F vs. c). For the first two smoothing methods the value of lambda is the same for all fields, for the third one the value varies across fields.

Determining Score Contribution for BM25, LM and DFR

Note that in Definitions 5.15, 5.16 and 5.17 $n(\mathrm{t}_{\mathrm{ra}}, f) = 1$ and $n(\mathrm{t}_{\mathrm{co}}, f) = 1$. This is an important point as it means that for the $\mathrm{S}_{\mathrm{contr}}$ components the score is calculated as based on the first occurrence of a term in a document. From Definition 2.14, we can see that if $n(t,d) = 1$ then $\mathrm{S}_{\mathrm{contr,BM25}}(t, f_i, d) = \mathrm{IDF}(t, F_i)$, the same is true for our DFR baseline model. This is because the TF component equals 1 if $(t,d) = 1$[1]. For LM the definition of $\mathrm{S}_{\mathrm{contr}}$ is more complex. From Definition 2.16 we can see that if $n(t,d) = 1$ then $\mathrm{S}_{\mathrm{contr,LM}}(t, f_i, d) = \log\left(1 + \frac{(1-\lambda)\frac{1}{|d|}}{\lambda P(t|c)}\right)$. For LM this thesis uses the Dirichlet-based smoothing, meaning $\lambda = \frac{|d|}{|d|+\mu}$. Note that the background model can be estimated as the collection $P(t|c)$, or the collection field $P(t|F_i)$. To clarify, the following definitions are used for calculating the $\mathrm{S}_{\mathrm{contr,[G,GA,LA]}}$ for BM25 and LM:

$$\mathrm{S}_{\mathrm{contr,G,BM25\text{-}DFR}}(t, f_i, d) := \mathrm{TF}_{\mathrm{BM25},k_1,b}(t,d,c) \cdot \mathrm{IDF}(t,c) = \mathrm{IDF}(t,c) \tag{5.40}$$

$$\mathrm{S}_{\mathrm{contr,GA,BM25\text{-}DFR}}(t, f_i, d) := \mathrm{TF}_{\mathrm{BM25},k_1,b}(t,d,c) \cdot \mathrm{IDF}(t,c) = \mathrm{IDF}(t,c) \tag{5.41}$$

$$\mathrm{S}_{\mathrm{contr,LA,BM25\text{-}DFR}}(t, f_i, d) := \mathrm{TF}_{\mathrm{BM25},k_1,b}(t,d,c) \cdot \mathrm{IDF}(t,F_i) = \mathrm{IDF}(t,F_i) \tag{5.42}$$

$$\mathrm{S}_{\mathrm{contr,G,LM}}(t, f_i, d) := \log\left(1 + \frac{(1-\lambda)\frac{1}{|d|}}{\lambda P(t|c)}\right) \tag{5.43}$$

$$\mathrm{S}_{\mathrm{contr,GA,LM}}(t, f_i, d) := \log\left(1 + \frac{(1-\lambda)\frac{1}{|d|}}{\lambda P(t|c)}\right) \tag{5.44}$$

$$\mathrm{S}_{\mathrm{contr,LA,LM}}(t, f_i, d) := \log\left(1 + \frac{(1-\lambda)\frac{1}{|d|}}{\lambda P(t|F_i)}\right) \tag{5.45}$$

Assuming no document length normalization as in Theorem A.1, if $n(t,d) = 1 \Rightarrow \mathrm{TF}_{\mathrm{BM25},k_1,b}(t,d,c) = 1$. The assumption of $n(t,d) = 1$ is in fact unnecessary for the ICFW-BM25. This is because the $\mathrm{S}_{\mathrm{contr}}$ components are included in the lambda threshold calculation only as a ratio, which means the TF component of the equation is cancelled out. In general, due to the composition of ICFW, it is more straightforward to use it with the BM25. In the above equations this is clear from the fact that for BM25 document length does not enter the equation.

[1] It is also this aspect of the ICFW that can be seen to justify the definition of probabilities that we discussed in Chapter 3 (Definition 3.10) with respect to what term metrics are used. By focusing on document frequencies, rather than term frequencies when calculating term probabilities, the number of features that need to be considered when calculating $\lambda_{\text{TD-th}}$ is significantly reduced for BM25 and DFR

5.6 Optimizing the ICFW Model

Even though the focus in this chapter and the thesis is on non-optimised SDR models, sometimes training data is available and should therefore be used. In its current form Definition 5.5 does not offer many options for optimization, other than the parameters of the underlying model. Therefore we add an additional static field weight that is optimised together with the parameters of the underlying model:

Definition 5.18 (ICFW-Optimised)**.** *Let $w_{\text{stat},i}$ be a static field weight applied over the whole field.*

$$\begin{aligned}&\text{RSV}_{\text{ICFW-opt},\vec{\lambda},M,\vec{w}_{\text{stat}}}(d,q,c) := \\ &\quad \sum_{i=1}^{m} w_{\text{stat},i} \cdot w_{\text{icfw},\lambda_i}(f_i,F_i,d,q) \sum_{t \in q} \text{RSV}_M(q,f_i,c)\end{aligned} \tag{5.46}$$

When training the model Definition 5.18 optimises the underlying model parameters (e.g. k_1 and b in for BM25), the static field weights and the lambda parameter. Two methods for optimising lambda are considered:

Definition 5.19 (ICFW-Lambda-Const (ICFW-LC))**.** *Let lambda be set as a constant for each field, meaning lambda is a vector of length m.*

$$\lambda_{\text{ICFW-LC}} := [\lambda_1 \ldots \lambda_m] \tag{5.47}$$

Definition 5.20 (ICFW-Lambda-Est (ICFW-LE))**.** *Let lambda be estimated in a linear regression manner from the mean and average of the queries IDF values:*

$$\lambda_{\text{ICFW-LE}}(q,c) := B_0 + B_1 \operatorname{mean}(\text{IDF}(q,c)) + B_2 \operatorname{var}(\text{IDF}(q,c)) \tag{5.48}$$

The focus of this thesis is on analytical models and potential standard models for SDR. This means that more emphasis is given to model candidates which do not need to be optimised. As a result of this, the time spent on optimised versions of ICFW is significantly smaller than on non-optimised versions. Further study into how to best optimise ICFW is left for the future.

5.7 Evaluation and Analysis

The purpose of this section is to demonstrate the effectiveness of ICFW using established benchmark datasets. After introducing the datasets the following research questions (RQs) will be answered.

RQ1: How well does ICFW do overall for the non-optimised task?

RQ2: What levels of lambda do we see?

RQ3: Which saturation method is best?

RQ4: How does saturating term frequency affect ICFW performance?

RQ5: How well can lambda be estimated analytically?

RQ6: Is the good performance of ICFW due to it satisfying the SDR constraints more comprehensively?

RQ7: How well did ICFW do overall for the optimised task?

The focus in the experimentation is on ICFW where the underlying model is the BM25. This is because the aim of the section is to demonstrate the value of ICFW as a field weighting method and it therefore makes sense to compare it to other strong aggregation methods with the same underlying model. The BM25F is the most established analytical SDR method, so the experimentation will focus on SDR models that use BM25 as the underlying model. Even though MLM represents a similar approach in the LM sphere of IR, not much time will be spent on it, or an LM-based version of ICFW. This is because MLM is not as established, or robust as the BM25F.

5.7.1 Data Collections

The experiments are performed on three test collections reflecting different document structure types from the more simple {*title*, *body*} of trec-web-small to the more complex {*names*, *related categories*, *similar entity names*, *entity name* and *attributes*} of DBpedia. The collections are DBpedia[2] [94], HomeDepot[3] and Trec-8-Small-Web[4]. The sizes of the data sets vary between 4.6 million (DBpedia) and 50k Homedepot. More important than the number of documents, is the complexity of the structure, as this shows that ICFW is robust across different structure types. For more information on the data sets see the footnotes.

All the collections were preprocessed with Krovetz stemming and by removing the standard English stopwords. The experimentation was conducted as a reranking task, with the initial retrieval done using ElasticSearch. All the learning is done using coordinate ascent (CA), optimizing for NDCG with 5-fold cross validation [106]. For ease of reproducibility, the implementation is

[2] https://github.com/iai-group/DBpedia-Entity

[3] https://www.kaggle.com/c/home-depot-product-search-relevance

[4] https://trec.nist.gov/data/t8.web.html

	DBpedia	HomeDepot	TREC-Web
Number of Documents	4.6M	55K	220K
Number of Queries	467	1000	50
Number of Rel. Judg.	49K	12K	47K
Number of Fields	5	3	2

Table 5.2: Test collection information

made available on GitHub at https://github.com/TuomasKetola/icfw-for-SDR .

5.7.2 Baselines

It is not our aim to demonstrate that ICFW outperforms all SDR models. Instead, we wish to show that ICFW is able to leverage the structure of the data in ways that existing analytical models are not. As we are comparing ICFW with existing field weighting methods, rather than existing SDR models, the experimentation will not include all the models in Chapter. 2. Instead, we will focus on the BM25 and LM retrieval models and their various fielded versions.

BM25

FSA-BM25: Linear sum of BM25 scores. **BM25F-Simple**: A BM25 model where document length normalization is applied over the concatenated document [69]. **BM25F**: Fielded BM25 model where document length normalization is applied at a field level [70]. For the non-optimised retrieval task, the BM25 parameters are set as $b = 0.8$ and $k_1 = 1.6$ (midpoint in the recommended range [1.2-2.0] [19]) and field weights are uniform. For the optimised task, models are optimised using coordinate ascent and 5-fold cross validation [106] for NDCG@100. The underlying BM25 model for all the approaches is the original one by Robertson et al. [107, 33].

Language Modelling

FSA-LM: Linear sum of LM scores using Dirichlet smoothing. **MLM**: Mixture of language models [71]. For the non-optimised retrieval task the Dirichlet hyperparameter is set as $2 \times$ avgfl and field weights are uniform.

Divergence From Randomness

FSA-DFR: Linear sum of DFR scores with TF-IDF basic model, Laplace-based first normalization and H1 second normalization. **DFR-F:** A fielded version of

DFR, inspired by the BM25F and MLM.

Definition 5.21 ($\mathrm{RSV}_{\text{DFR-F}}$). *Let $n_{\vec{w}}(t,d)$ be the weighted sum of term frequencies over the fields, $n_{\mathrm{norm}}(t,f_i) := n(t,d) \cdot \frac{\mathrm{avgfl}}{|f_i|}$ and $n_{\mathrm{norm},\vec{w}}(t,d) = \sum_i^m w_i n_{\mathrm{norm}}(t,f_i)$.*

$$\begin{aligned} &\mathrm{RSV}_{\text{DFR-F}}(d,q,c) \\ &\quad := \sum_{t \in q} \mathrm{Inf}_1(t,d,\text{TF-IDF}) \cdot \mathrm{Inf}_2(t,d,L) && (5.49) \\ &\quad = n_{\mathrm{norm},\vec{w}}(t,d) \cdot \log \frac{N+1}{\mathrm{df}(t,F_i)+0.5} \cdot \frac{1}{1+n_{\mathrm{norm},\vec{w}}(t,d)} && (5.50) \end{aligned}$$

Catchall Field

For the FSA-based models (baselines and candidate models) we consider model versions where a catchall field has been appended to the data collection. All the other fields have simply been flattened into this single field, meaning it is a non-structured representation of the collection. This has been done as it is an easy method for getting rid of some of the noise for FSA-based models resulting from the independence assumption between term occurrences across fields. The consideration of the catchall field is denoted by an addition of +all to the model name.

5.7.3 Candidate Models

The experimentation focuses on testing how the satisfaction of the TD-Constraint affects ICFW performance. Satisfying the FD-Constraint is more straightforward as it is only concerned with term-specific variations in the specificity (IDF) values. Furthermore, from Chapter 4 we know that there are more performance gains to be made by a model satisfying the TD-Constraint than the FD-constraint. The fact that TFA-based models (BM25F and MLM) are considered more robust than their FSA counterparts is also testament to this. Focusing on the TD-Constraint means that we assume no variation for the specificity of a given term across fields in the experimentation.

For the above reasons the candidate models set lambda according to the TD-Constraint threshold, that is according to Definition 5.9. We consider three different candidates for setting lambda related to the three smoothing techniques from Section 5.5, together with assuming $x = 1$ (the DFR has the same definition for $\mathrm{S}_{\mathrm{contr}}$ as BM25). As discussed numerous times $x = 1$ is not a realistic assumption in a real-world retrieval scenario, but the above-described focus on variation of specificity across terms, rather than across fields for a given term justifies it. This does not mean that future methods could not account for it as well. The discussion regarding the satisfaction of FD-Constraint in Section 5.4.2 provides a good starting point for this analysis.

To summarise the model candidates for ICFW are: **ICFW-G** where Definition 5.15 is used to estimate the underlying metrics, **ICFW-GA** where Definition 5.16 is used and **ICFW-LA** where Definition 5.17 is used. For each model candidate, we also consider a version with the catchall field. For the supervised models we consider **ICFW-LC** and **ICFW-LE** for the two lambda optimization methods from Section 5.6.

The results also show the accuracy of ICFW if $\lambda = 0$, meaning the model is effectively the same as the BM25-FIC (denoted **ICFW-λ-zero**). Some figures also consider a semi-optimised version of ICFW where a single value for lambda is optimised, it is denoted **ICFW-const-λ**. Finally, **ICFW-best** denotes the best-performing ICFW candidate.

5.7.4 Measuring Significance

Significance tests has been applied, even though there are different views on the methodology. Fuhr and Sakai [108, 109] make the case that significance tests should not be used on multiple hypotheses (without correction) and that simple significance tests should not be applied to re-used test collections. Though the authors share similar views, significance tests are still often considered a must-have. Furthermore, some test collections are not reused (Homedepot) and the proposed models are similar, meaning as features they are correlated.

5.7.5 Answering the Research Questions

RQ1: How well did ICFW do overall for the non-optimised task?

BM25

Table 5.3 shows the overall results for the experimentation. As discussed earlier in the chapter, the results have been separated into non-optimised and optimised tasks. For the non-optimised task, we report the performance of ICFW-λ-zero where $\lambda = 0$ (i.e. BM25-FIC) and for each of the smoothing methods from Section 5.5 we report the accuracy both for when a catchall field is considered (-all) and when it is not.

Overall it is clear that ICFW does better than the best-performing baseline for each data collection. Furthermore, this improvement is statistically significant at $p < 0.01$ for most proposed models. Figure 5.9 puts the scale of this improvement in context.

From the figure can see that using ICFW for field weighting instead of optimising BM25F provides approximately half the performance gains. This should

dataset	dbpedia		trec-web		homedepot	
metric	map	ndcg@100	map	ndcg@100	map	ndcg@100
Non-optimized						
Baseline Models						
FSA-BM25	0.226	0.351	0.164	0.290	0.252	0.452
BM25F	0.295	0.444	0.229	0.377	0.249	0.440
BM25F-Simple	0.284	0.433	0.229	0.378	0.238	0.429
FSA-BM25+all	0.256	0.393	0.205	0.349	0.258	0.458
Proposed Models						
ICFW-λ-zero	0.207	0.331	0.225	0.366	0.297	0.496
ICFW-G	0.299	0.448	0.243	0.391*	0.290*	0.486*
ICFW-GA	0.302	0.449	0.241	0.389	0.297*	0.496*
ICFW-LA	0.304*	0.453	0.233	0.378	**0.299***	**0.498***
ICFW-λ-zero-all	0.239	0.369	0.245	0.395	0.290	0.488
ICFW-G+all	0.305*	0.459*	**0.251***	**0.406***	0.277*	0.470*
ICFW-GA+all	**0.313***	**0.468***	0.249*	0.403*	0.285*	0.482*
ICFW-LA+all	0.310*	0.464*	0.249*	0.402*	0.289*	0.487*
Optimized						
Baseline Models						
FSA-BM25-CA	0.317	0.473	*0.286*	0.449	0.352	0.538
BM25F-CA	0.338	0.494	0.279	0.444	0.354	0.544
BM25F-Simple-CA	0.330	0.483	0.279	0.441	0.337	0.526
FSA-BM25+all-CA	0.334	0.492	0.286	**0.451**	0.358	0.547
Proposed Models						
ICFW-LC-CA	0.335	0.489	**0.287**	0.450	0.358	0.547
ICFW-LE-CA	0.336	0.489	0.285	0.448	0.356	0.545
ICFW-LC+all-CA	**0.344***	**0.500**	0.281	0.442	0.358	0.547
ICFW-LE+all-CA	**0.344***	0.499	0.280	0.441	**0.360**	**0.548**

Table 5.3: Experimentation results with BM25 as the underlying model ($M =$ BM25 in Definition 5.5). The percentages show the increase compared to the best performing baseline. * denotes significance at $p < 0.05$ for a Wilcoxon signed ranks test. +all means the model considered a catch_all field with all fields concatenated.

be considered a notable improvement. Optimising BM25F uses training data to understand the importance of fields and set their weights as well as the correct hyperparameters (k_1 and b). This means that the model has been built for a specific dataset and a set of queries, which takes time and effort; if it is even possible. Furthermore, there is no guarantee that it generalises to other queries and data sets. Amongst others [94] demonstrated that the optimal values for k_1 and b can change drastically if a different set of queries is considered with the same data. On the other hand, ICFW does not require any training and can be used off the shelf. This makes the relative improvement of using ICFW for field weighting large compared to optimising BM25F, even though the latter is more accurate.

LM

Table 5.4 shows the experimental results if LM is used as the underlying model. Compared to BM25, the first observation is that the LM baselines do much worse, especially for Trec-Web. A potential reason for this is that the field-based scores for FSA-LM are not necessarily very comparable. The differ-

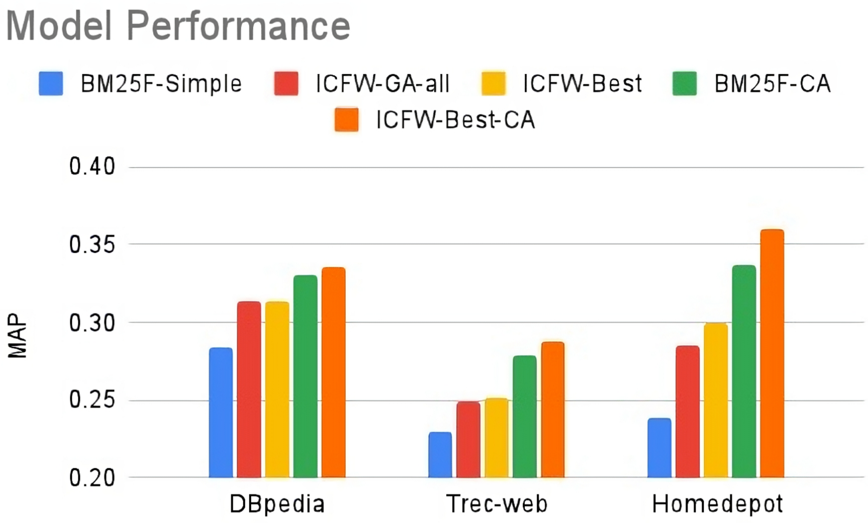


Figure 5.9: The scale of performance improvement provided by the ICFW put into context. The accuracy difference between ICFW candidates, a non-optimised BM25F and an optimised BM25F. -CA denotes coordinate ascent-based optimization

ence in document length between the title and body is significant and affects the retrieval score directly, resulting in a situation where the title scores are inherently smaller. Hence some form of normalization might be called for. This, however, is outside of the scope of this experimentation. A notable exception to this is the Homedepot data collection where LM performs much better than BM25.

In a similar manner to the BM25 analysis, Figure 5.10 puts into context the improvements gained from using ICFW. For two of the data collections (DBpedia, Homedepot) there is in fact a higher increase in accuracy from using ICFW than from optimizing MLM. This is notable as it suggests that if the underlying model is LM, ICFW can provide higher performance gains without any optimization than baselines can with optimization.

For Trec-Web the results are problematic at best due to the field length issue discussed above. These issues are likely to affect the ICFW models as well. How to best solve them is left to future research. The rest of the experimentation focuses on ICFW-BM25.

DFR

Table 5.5 shows the experimental results if DFR is used as the underlying model. The trend is similar to the BM25-based experimentation. A notable exception is that the ICFW-λ-zero-all model is the best performing one for

dataset	dbpedia		trec-web		homedepot	
metric	map	ndcg@100	map	ndcg@100	map	ndcg@100
Non-optimized						
Baseline Models						
FSA-LM	0.201	0.323	0.093	0.151	0.282	0.472
MLM	0.232	0.345	0.094	0.151	0.300	0.480
FSA-LM+all	0.213	0.339	0.101	0.168	0.263	0.454
Proposed Models						
ICFW-λ-zero	0.260	0.388	0.103	0.162	0.327	0.513
ICFW-G	0.272	0.407	0.105	0.173	0.318	0.499
ICFW-GA	0.271	0.405	0.106	0.172	0.325	0.507
ICFW-LA	0.273	0.407	0.103	0.164	**0.328**	**0.513**
ICFW-λ-zero-all	0.274	0.409	0.119	0.194	0.323	0.510
ICFW-G+all	0.279	0.418	0.120	0.199	0.316	0.500
ICFW-GA+all	0.282	0.421	0.121	0.197	0.322	0.506
ICFW-LA+all	**0.284**	**0.422**	**0.122**	**0.200**	0.323	0.510
Optimized						
Baseline Models						
FSA-LM	0.273	0.416	0.218	0.356	0.304	0.482
MLM	0.275	0.414	0.239	0.392	0.306	0.484
FSA-LM+all	0.287	0.434	0.218	0.356	0.304	0.483
Proposed Models						
ICFW-LC	0.310	0.459	0.238	0.388	0.308	0.488
ICFW-LE	0.306	0.455	0.238	**0.391**	0.313	0.493
ICFW-LC+all	**0.315**	0.466	**0.239**	**0.391**	0.311	0.490
ICFW-LE+all	0.314	**0.467**	0.236	0.384	**0.319**	**0.500**

Table 5.4: Experimentation results with LM as the underlying model (M = LM in Definition 5.5). The percentages show the increase compared to the best performing baseline. * denotes significance at $p < 0.05$ for a Wilcoxon signed ranks test. +all means the model considered a catch_all field with all fields concatenated.

DBpedia in the non-optimised task.

Figure 5.11 simplifies Table 5.5 so that we can easily see the degree to which ICFW increases performance in both tasks. It is noteworthy that the two ICFW models perform as well as the optimised DFR-F model for both DBpedia and Trec-Web. This means that we observe higher accuracy gains for a non-optimised retrieval model, than an optimised one with the same underlying retrieval function (DFR). A possible reason for this is the fact that, unlike BM25F, DFR lacks the hyperparameters for adjusting term frequency saturation and document length normalization, both at the field level and at the document level.

RQ2: What kind of lambda levels do we see?

Figure 5.12 shows the distribution of lambda for the different data collection and model candidates with BM25 as the underlying model. These are the lambdas that produced Table 5.3 From the figure, we can see that $\lambda < 8$ for all data collection and model candidate pairs, with higher values being significantly less likely. This finding bodes well for ICFW the satisfaction of the FD-Constraint

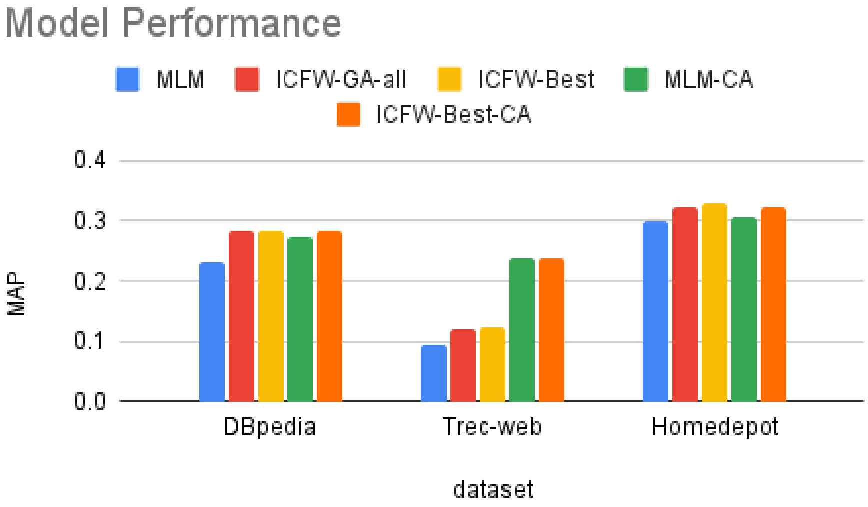


Figure 5.10: The accuracy difference between ICFW candidates, a non-optimised MLM and an optimised MLM. -CA denotes coordinate ascent-based optimization

as if we assume that $x = 1$, $T = z$ and $k_1 = 1.6$, meaning $\zeta = 1.44$ the lambda-threshold for satisfying the FD-Constraint ($\lambda_{\text{FD-th}}$) is equal to 7.9, meaning the constraint is satisfied for almost all queries in our data-collection.

RQ3: Which saturation method is best?

None of the smoothing methods from Sections 5.5 clearly outperforms the others for all data collections. In fact, each data collection has a different best-performing method. For clarity of the rest of the analysis, it is worth choosing a single smoothing method so that the discussion can be generalised to all the data collections in question. ICFW-GA+all is the best compromise out of the six proposed models.

RQ4: How does saturating term frequency affect ICFW performance?

Figure 5.13 shows the effect of lambda, i.e. cross field term frequency saturation on NDCG@100. The underlying model is BM25 (M = BM25 in Equation (5.8)). For RQ1 it is worth only considering the top three graphs, as the bottom ones have inherent saturation due to the catchall field. ICFW-const-λ shows the accuracy for an ICFW model where a single value for lambda is optimised. At $\lambda = 0$, the ICFW-const-λ model corresponds to the BM25-FIC [1]. We can see that for all the datasets (without catchall field), the optimal value of λ is greater than 0, albeit for Homedepot even at $\lambda = 0$, ICFW outperforms

dataset	dbpedia		trec-web		homedepot	
metric	map	ndcg@100	map	ndcg@100	map	ndcg@100
Non-optimized						
Baseline Models						
FSA-DFR	0.216	0.344	0.152	0.271	0.256	0.455
DFR-F	0.267	0.411	0.208	0.342	0.249	0.439
FSA-DFR+all	0.250	0.386	0.192	0.329	0.263	0.463
Proposed Models						
ICFW-λ-zero	0.288	0.430	0.212	0.346	0.303	0.502
ICFW-G	0.294	0.441	0.229	0.370	0.298	0.493
ICFW-GA	0.296	0.441	0.227	0.366	0.305	0.503
ICFW-LA	0.300	0.447	0.218	0.358	**0.305**	**0.504**
ICFW-λ-zero-all	**0.314**	**0.462**	0.231	0.376	0.296	0.494
ICFW-G+all	0.298	0.450	0.237	**0.386**	0.286	0.478
ICFW-GA+all	0.308	0.460	0.234	0.383	0.294	0.491
ICFW-LA+all	0.303	0.455	0.233	0.380	0.297	0.494
Optimized						
Baseline Models						
FSA-DFR	0.270	0.409	0.211	0.348	0.335	0.526
DFR-F	0.308	0.454	0.233	0.377	0.345	0.532
FSA-DFR+all	0.299	0.444	0.212	0.350	0.345	0.537
Proposed Models						
ICFW-LC	0.311	0.459	**0.235**	**0.383**	0.357	0.544
ICFW-LE	0.310	0.458	0.232	0.378	0.356	0.544
ICFW-LC+all	0.321	0.472	0.234	0.382	0.357	0.544
ICFW-LE+all	**0.323**	**0.474**	0.233	0.380	**0.357**	**0.547**

Table 5.5: Experimentation results with DFR as the underlying model ($M =$ DFR in Definition 5.5). The percentages show the increase compared to the best performing baseline. * denotes significance at $p < 0.05$ for a Wilcoxon signed ranks test. +all means the model considered a catch_all field with all fields concatenated.

baselines clearly. This explains the results in [1]. So we can conclude that saturating term frequency across fields is indeed important. It is likely to be more important for data structures such as {title, body}, where there is greater dependence between term occurrences. This is evident in Figure 5.13 from the high value of optimal lambda for Trec-Web.

RQ5: How well can lambda be estimated analytically?

Figure 5.13 demonstrates that lambda can be estimated well analytically. We observe that the best performing smoothing method for each dataset from Sec. 5.5 (ICFW-best) is able to locate the maximum point of ICFW-const-λ well, even for very different values of lambda (0 for Homedepot+catchall vs. 16.5 for Trec-Web). ICFW-best is different for all the datasets: ICFW-LA, ICFW-G and ICFW-LA for DBpedia, Trec-Web and Homedepot respectively with no catchall field and ICFW-GA-all, ICFW-G-all and ICFW-LA-all with the catchall field. Out of the three smoothing methods for df, ICFW-G is the most straightforward one and its performance is therefore also reported in Figure 5.13. It only falls significantly short of ICFW-best for Homedepot.

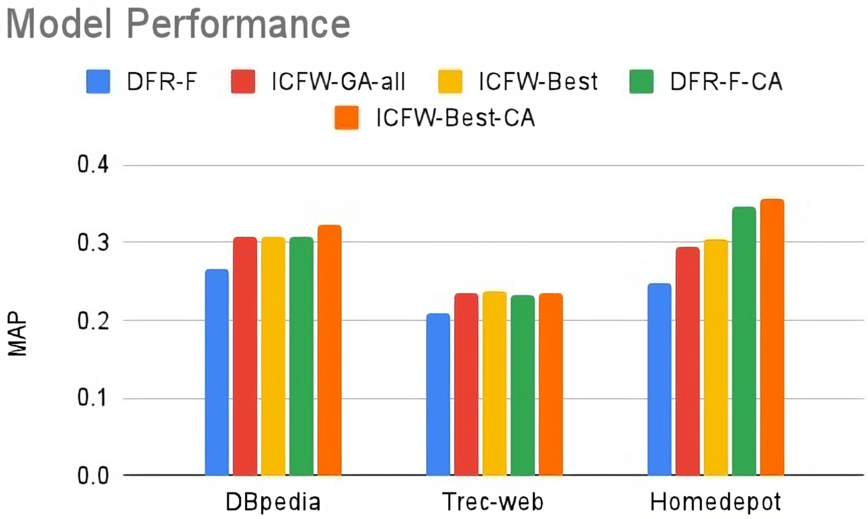


Figure 5.11: The accuracy difference between ICFW candidates, a non-optimised DFR-F and an optimised DFR-F. -CA denotes coordinate ascent-based optimization

RQ6: Is the good performance of ICFW due to it satisfying the SDR constraints?

Table 5.6 suggests the reasons why ICFW outperforms baselines. The analysis was done for BM25-based models on the HomeDepot data collection where the number of queries is the highest. This was done to make the results less noisy. By calculating the correlation of three query features with MAP difference between ICFW and the baseline models, we can see that ICFW behaves as we would expect given its grounding in the constraints from Chapter 4. A higher average proportion of fields query terms appear in, is associated with a larger accuracy increase for ICFW compared to both the BM25F and BM25-FSA. This makes sense as terms appearing in many fields make the models more prone to

	ΔMAP (BM25F → ICFW)	ΔMAP (FSA-BM25 → ICFW)
Field Proportion	+0.220†	+0.196†
Term Proportion	+0.105†	+0.284†
max(IDF) − min(IDF)	+0.05	+0.12†

Table 5.6: Query feature correlation analysis on Homedepot data set (n=1000). Field proportion = average proportion of fields a query term appears in. Term proportion = average proportion of query terms in a document. Only relevant documents are considered. † dagger denotes significance at 0.01 p-value.

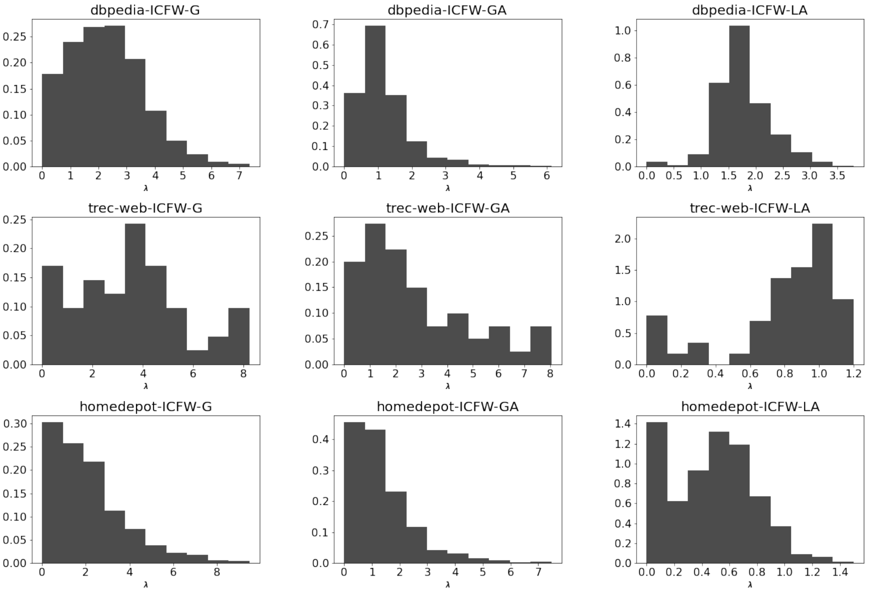


Figure 5.12: Histograms of lambda levels observed in experimentation

problems caused by not satisfying the SDR constraints. The same is true for the second query feature in Tab 5.6: If relevant documents are more likely to have most of the query terms, their dependence across the fields becomes more important. As BM25-FSA does not account for this, it loses on accuracy. BM25F is less affected by many query terms appearing as opposed to few, as term frequencies are saturated across fields. However, the problem discussed in Theorem A.1 is still present. The results in Table 5.6 are consistent with this. A larger difference in the IDFs of the terms can also create problems. The issue is only significant for FSA as term frequencies are not saturated across fields.

The correlation analysis demonstrates how query features that make TFA and FSA vulnerable to problems associated with failure to satisfy the constraints from Sec. 4 are correlated with how well the models perform against ICFW. This suggests that the increased performance for ICFW is tied to constraint satisfaction.

RQ7: How well did ICFW do overall for the optimised task

First considering the results for ICFW with BM25 as the underlying model. The lower half of Table 5.3 presents the results for the supervised task. It is clear from the results that the differences between the retrieval accuracy of the models, both between the baselines and the ICFW models, are much smaller. None of the baselines is clearly better than others across data collections. Interestingly, BM25F — which is usually considered state of the art for analytical SDR models

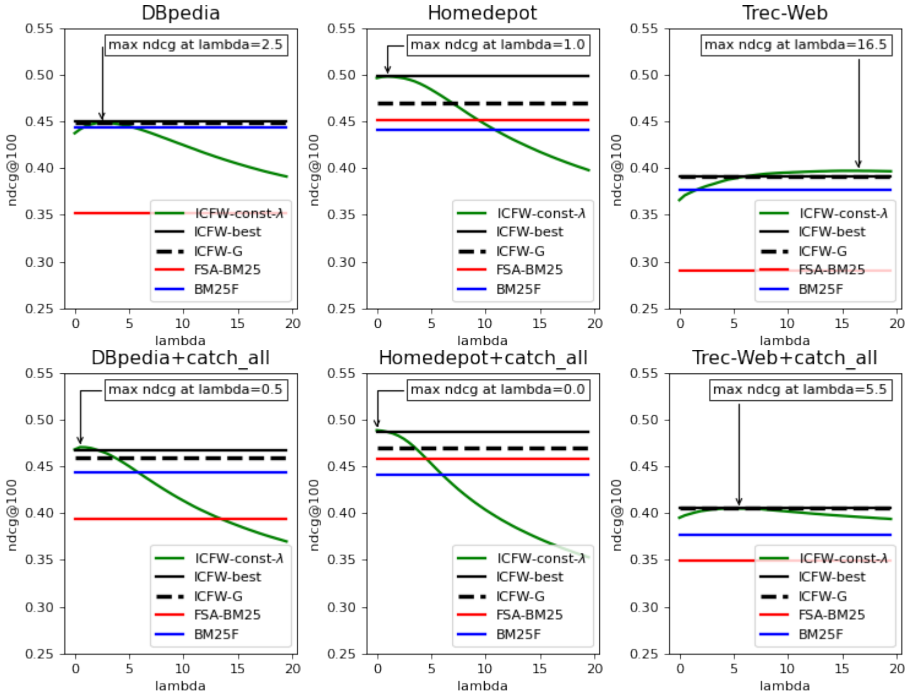


Figure 5.13: Estimating lambda vs. optimising lambda

— is the best only for DBpedia. Even there, the difference to FSA-BM25+all is marginal. There are a few cases where we observe significant differences in retrieval behaviour between the baselines and ICFW. In general, the findings seem to correspond to the observation made by [110] for non-structured retrieval, that once hyperparameter optimization is used, bag-of-words-based analytical models do not differ significantly in accuracy.

The results are notably different if the underlying model is LM compared to BM25. We see a much bigger improvement for ICFW compared to the baseline models. Only for Trec-Web, there is no significant increase in accuracy.

5.7.6 Discussion

The experimentation has demonstrated that ICFW used together with BM25 clearly outperforms the baseline models FSA-BM25 and BM25F in a retrieval scenario without training data (non-optimised). The importance of smoothing the lambda value estimated by Definition 5.9 was demonstrated, though none of the smoothing methods clearly outperformed the others across data collections. ICFW-GA-all was deemed to be a good compromise, with large performance increases for all data collections compared to the best baseline models. The experimentation further demonstrated that the reason for the large increases in accuracy for the non-optimised task was due to ICFW saturating term frequency across fields and therefore satisfying more of the SDR constraints from Chapter 4 than other existing methods. The theoretic foundation of ICFW, coupled with its satisfaction of the SDR constraints and the demonstrated performance on established benchmarks make it a good potential candidate for a new standard SDR method.

5.8 Summary, Conclusions and Contributions

This chapter covered the following issues:

- The introduction and formalization of the ICFW method for field weighting, with term frequency saturation.
- Discussion and demonstration of the importance of setting the scale parameter lambda in order for ICFW to satisfy the SDR constraints from the previous chapter.
- Analysing the importance of smoothing lambda.
- Extensive experimentation on three benchmark data collections.

The main conclusions were:

- By using adding the document-based information content and the lambda scaling feature to the (BM25-)FIC model term frequency can be saturated across fields, which results in the model conditionally satisfying all four retrieval constraints from Chapter 4.
- Smoothing lambda is important and has an effect on performance.
- The resulting models with various smoothing methods outperform existing SDR models by a large margin, especially when it comes to retrieval scenarios where the models cannot be optimised using training data.
- ICFW is a feasible candidate for becoming a reliable standard for analytical SDR.

The main contributions are:

- ICFW; a method for using information content for field weighting where term frequency can be saturated across fields.
- Theoretic foundation for ICFW.
- Analysis of how ICFW satisfies SDR constraints formulated in Chapter 4.

- Formal evaluation of model on established benchmarks demonstrating how ICFW outperforms existing SDR models by a large margin, especially for a non-optimised retrieval scenario.

Chapter 6

Relevance Structure-based Entity Ranking and Investigative Information Retrieval (InvIR)

This section introduces the **R**elevance **S**tructure-based **E**ntity **R**anking (RSER) system. It is a prototype search engine for InvIR which uses the ICFW to help users understand the structure of the data better and to rank interesting entities. The aim of the chapter is to demonstrate the importance of ICFW in an InvIR task where document structure plays an important part. It will be demonstrated that the ICFW weights can be used by the user to better navigate during the investigative process and that they can be used by the system to consider what we call relevance structures; defined as different contexts in which entities occur in the data. The chapter is structured as follows:

- Section 6.1 briefly describes the motivation for the chapter.
- Section 6.2 introduces the proposed search engine.
- Section 6.3 summarises relevant research.
- Section 6.4 introduces the concept of relevance structures.
- Section 6.5 describes the technical aspects of the engine.
- Section 6.7 explains the implementation and evaluation.
- Section 6.8 concludes.

6.1 Motivation

As discussed in the Introduction (Chapter 1), the initial motivation for this thesis came from helping investigators dig through mountains of unruly data. Whether it is investigative journalists, law enforcement, or open source investigations, the professionals working in these areas require systems with a high degree of transparency in order for them to trust their findings. More recently much of the research in search and retrieval has focused on black-box learning algorithms and large language models. These methods are powerful and have been shown to outperform traditional approaches. However, as has been pointed out throughout this thesis, they lack the transparency required for direct use in investigations. At the very least they must be complemented by more transparent analytical models to ensure that the findings are valid and truthful. Developing these methods has been the overarching aim of this thesis.

The leveraging of document structures was identified as the technical method for developing transparent and interpretable retrieval models. The last three chapters have presented these models, which have been shown to outperform existing SDR models.

The thesis now turns to answering the original question of "How to help investigators?" from a more practical and application-driven perspective, using the technical tools developed in the previous chapters. The aim of this chapter is to present and evaluate a prototype investigative search system, which — if developed further — could be used by investigators with no prior knowledge of the structure of the data to rank interesting entities. The intention here is not to "solve" or "automate" the task of InvIR, but to demonstrate how the retrieval methods introduced in this thesis can help with this task. As a result of this, more attention is paid to the description of the system and its underlying idea, rather than formal evaluation.

6.2 Introduction

This section first introduces the retrieval task at hand, followed by the proposed method for solving said task.

6.2.1 Retrieval Task

As discussed in the Background chapter (Chapter 2), search is a central aspect of data-driven investigations, where investigators dig through data collections for previously unknown facts. These users can be journalists exploring public data and leaks[1][2], law enforcement offices investigating data obtained through foreclosures, or open source investigators scouring social media data for evidence of dubious activity etc. [111, 112]. In all the scenarios above, it is possible that the investigators have a list of "Entities of Interest" (EoIs) that they think could be found in the data. This is because such investigations usually aim to find people of importance, such as politicians in these data collections. Furthermore, they might already know of an interesting entity in the data, this seed entity (SE) can be used as a reference point.

The retrieval task considered is as follows: Given a user's information need, a list of EoIs, the user's knowledge of a seed entity and their specific interest in the seed entity, rank the list of EoIs based on whether they can be found in a similar context as the seed entity in the data.

As a more concrete example consider the following: The information need is "List of Russian people that keep money in tax heavens, own a yacht and have ties to the government?", the seed entity SE could be Arkady Rotenberg, who is known to satisfy the information need well. The list of EoIs could be every influential person in Russia for example (n=10k+). Given this information, we would like to rank the EoIs based on whether they can be found in a similar context in the data as Arkandy Rotenberg, i.e. whether they have money in tax heavens, own a yacht and have ties to the government. This would significantly ease the work of the investigator, as they would have a better idea of which entities they should start with.

It is imperative that a user can easily understand the inner workings of the system in terms of why it produces the ranking it does. Otherwise, the investigator cannot trust the system. This is why so much emphasis is given to the transparency of the system, through the transparency and analytical nature of the underlying models, as well as the UI.

The above example describes the motivation for the proposed system well in the context of investigations and the reduction of labour for the investigator. However, its complexity makes it difficult to clearly explain the inner workings of the system. For this purpose, it is easier to consider an example with movie-related data. Table 6.1 demonstrates how the engine defines the context and how the EoIs are ranked for data about movies, actors and characters.

The list of entities contains actors and characters from movies about magic: Malfoy, Bilbo Baggins, Alladin, Coulter, Tom Felton, Nicole Kidman, Martin Freeman, and Robin Williams. Consider two information needs, the first one is actors in movies about magic and the second one is characters in movies about magic. The information need is described by the base query "wizards magic

[1] https://aleph.occrp.org/

[2] https://datashare.icij.org/

rank	ranking for SE_1	ranking for SE_2
1	Tom Felton	Malfoy
2	Nicole Kidman	Bilbo Baggins
3	Robin Williams	Alladin
4	Martin Freeman	Coulter
5	Coulter	Tom Felton
6	Malfoy	Nicole Kidman
7	Alladin	Martin Freeman
8	Bilbo Baggins	Robin Williams

Table 6.1: Example entity ranking scenario where a user is looking to rank entities in movies about magic based on context. q = *wizards magic fantasy.* The first context is actors, second context is characters. SE = seed entity. SE_1 = Emma Watson, SE_2 = Hermione.

fantasy" and a seed entity that is chosen by the user. For the first information need the user chooses Emma Watson who they know is an actor in a magic movie (Harry Potter) and for the second one they choose Hermione as they know she is a character. Given this information, the system, with the help of the user, should produce the two rankings of EoIs in Table 6.1 corresponding to the two contexts.

6.2.2 Proposed Method

The core contribution of this chapter is to introduce the **I**nvestigative Search for **R**elevance **S**tructure-based **E**ntity **R**anking (RSER) system. The algorithms used are transparent and the reasoning behind the final ranking is easily available to the user, meaning the user can trust the results more. As discussed in Chapter 2, this is an important aspect of investigative search.

Using the example in Table 6.1 the system works as follows: The area of interest for the information need is defined through a set of queries Q, in Table 6.1 this is a single query q = *wizards magic fantasy.* The seed entity SE is appended to the base query and an initial search is performed ($q(\text{SE}_{\text{Emma Watson}})$ = wizards magic fantasy Emma Watson). For both SE_1 and SE_2, the user chooses interesting documents which provide evidence of the seed entity's desired context, in this case, the Harry Potter movies. The discovery of these interesting documents is made easier by a user interface which helps the user navigate and learn about the document structure. How the interface accomplishes this is an important part of the contributions of this chapter.

Using the interesting documents found, the system ranks EoIs based on how well their context in the data matches that of the seed entity. This context is estimated using relevance structure of each entity, i.e. the ways in which the document structure affects their relevance. So if SE = Emma Watson (context

= actors) the top ranks are actors in movies and if SE = Hermione (context = characters) they are characters.

This simple example demonstrates how a user can easily navigate a structured data collection and learn new information about EoIs using a known seed entity. A real-world investigative scenario could be much more complicated in terms of the underlying data structure, the base queries used and the EoI list. As discussed in the previous section, the data collection could be a set of documents about companies and their ties to individuals and politicians, the base queries could produce documents demonstrating corruption and the EoIs could be persons of influence for a given country. After interaction and exploration, the system would give a ranking of the EoIs that are most likely to be relevant to the investigator. It is easy for the user to find out and understand why an entity gets a high rank, meaning the user can reason with the system to understand better the why behind it all.

As mentioned before, the focus of this chapter is on the description of the proposed system and its underlying idea. However, a small-scale formal evaluation is performed as well. RSER is evaluated on two test collections created from movie data collected from IMDB by Bamman et al. [113] and DBpedia, which is introduced in previous chapters. The test collections has been modified to fit the retrieval scenario above. All together 25 topics have been created. Different initial retrieval models are compared, as well as different models for comparing the similarity of EoI's the seed entities' relevance structures.

6.3 Background and Context

The purpose of this section is to clarify the context of the proposed investigative search engine with respect to existing SDR research, including ICFW and with respect to polyrepresentaiton.

6.3.1 Existing SDR Models suitable for RSER

As the structure of the documents is a central aspect of the system we are introducing it is worth recapping briefly the existing approaches to SDR discussed in Chapter 2 and examining whether they are suitable for RSER. Wilkinson [74] was the first to show that leveraging document structures is beneficial for retrieval performance. Llamas offered a more theoretical approach to SDR using theory of evidence [75, 76]. The INEX initiative was big in the 2000s, with a lot of focus given to presentation and hierarchical data [114, 79, 80]. Fielded versions of established atomic retrieval models were introduced in [69] and [71] for the BM25 and language modelling respectively. More recently deep learning methods have been applied to SDR as well [30, 93].

In order for an SDR method to be viable for the system introduced here, it must differentiate between the document fields in terms of how they contribute to the relevance of the document, meaning TFA-based models are not viable as they do not satisfy the FD-Constraint, or the term importance constraint (TI-Co) from Chapter 4. As our intention is to develop an investigatory search system, another important model feature is transparency (As defined in Section 2.3), meaning deep learning-based black-box models are not feasible. Finally, the underlying assumption is that we are dealing with new data and a structure that we are not fully familiar with. This means that the SDR models we use are not optimised and must therefore work without training data. Furthermore, they cannot use the semantics of the field name as they are not known.

From all the SDR models mentioned in this thesis, we are then left with the FSA-based models and ICFW. In the previous chapter, we saw that ICFW-BM25 outperforms FSA-BM25, which means that it is likely that ICFW-BM25 is better for the proposed system as well, However, as we cannot be sure, the experimentation also includes runs where FSA-BM25 is used as the underlying model.

6.3.2 Polyrepresentation and Document Structures

The concept of relevance structure is an important aspect of the proposed investigative search engine and will introduced at length in the next section. It relates closely to the concept of polyrepresentation and the principle of polyrepresentation. First introduced by Peter Ingwersen based on the cognitive approach to IR, the principle of polyrepresentation states that given a set of information contexts (different sources, observers etc.), documents that exist in what is known as the cognitive overlap are most likely to be relevant [115]. In more concrete terms, Frommholz [116] describes the different contexts as different document fields, title, body text or review of a book in their example. Furthermore, they describe the interplay of these contexts during the retrieval process, i.e. query terms appearing in various document fields as different information needs. In their example for the query "A good introduction to quantum mechanics", the word "good" would need to appear in a book review field, rather than the body of the book in order for the query intent to be understood correctly. This means that information need and the parts of the document where query words appear are inherently connected. According to the principle of polyrepresentation the most relevant documents can be found where these information needs overlap [115].

Connecting the document structure and information need is done in a similar way in the proposed investigative search system: an information need is indeed characterized by how query terms and document fields are connected. Consider the query “Clint Eastwood” over a movie database: one possible information need is to find movies with Clint Eastwood as an actor and another as a director. However, unlike with the principle of polyrepresenation, the proposed system does not assume that the most relevant documents are those where the many fields overlap. Meaning a movie with Client Eastwood as the director and an actor is not necessarily considered to be the most relevant. Instead, we see each combination of fields as a separable information need: Movies with Clint Eastwood as both director and actor could represent just another information need, just like movies with him as only an actor, or director. Each of these three possible combinations of fields represents one relevance structure, which the proposed system helps the user navigate.

6.4 From Document Structure to Relevance Structure

So far the majority of this thesis has been concerned with document structures strictly in terms of how they can be leveraged in order to formulate models that perform better than their existing counterparts.. Document relevance has not been discussed at length and has really only been an underlying concept in the retrieval models considered and developed. There are many different formal definitions for relevance within the IR discipline. See the following works for a further discussion [117, 118, 119]. For the purposes of this chapter, a simple definition of document relevance can be adopted: relevant documents are defined as the documents the user finds useful/interesting.

Up to this point document structures have only been used to improve retrieval performance with respect to accuracy on benchmark collections. In the introduction chapter it was stated that a good investigative search system would also communicate information about the structure to the user, putting them in a better place to formulate further queries and providing a higher degree of transparency. There are limited benefits in simply describing the structure of a relevant document to the user; furthermore, this does not need to be done in conjunction with the retrieval model. However, what the RSER system does is to describe the structure of the document in terms of how it affects the relevance of the document. This is where the concept of relevance structures comes in.

A relevance structure describes how the structure of the document contributes to the system’s perceived relevance of a document with respect to a query, i.e. the retrieval score. Put in another way, relevance structure describes the composition of a document's relevance with respect to field relevance. Relevance structure can be visualised in many ways. Figure 6.1 demonstrates the use of a histogram, which is also used in the proposed system.

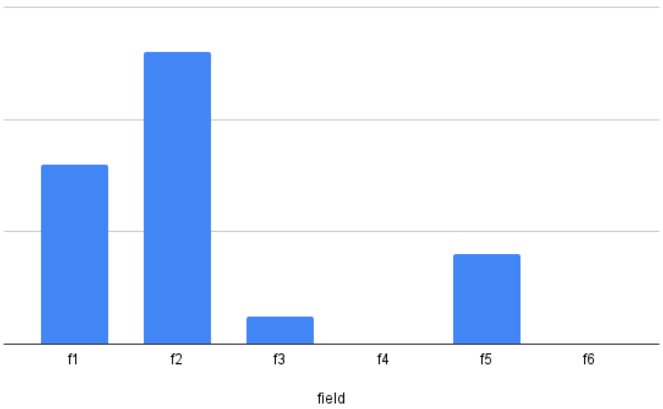


Figure 6.1: The relevance structure of document d ($d = [f_1 \ldots f_6)]$) presented as a histogram.

More formally, relevance structure is defined as follows:

Definition 6.1 (Relevance Structure)**.** *Let r_i be the relevance of a document field f_i. The relevance structure vector of a document with respect to query q and collection c is denoted $\vec{\text{rs}}$.*

$$\vec{\text{rs}}(q, d, c) := [r_1(q, f_1, c)...r_m(q, f_i, c)] \tag{6.1}$$

r_i can be defined in many different ways. The most naive method would be to define it as a field-based RSV: $r_i(q, f_i, c) := \text{RSV}_M(q, f_i, c)$. However, we have seen in the previous chapters that raw field-based scores do not model the relevance of document fields well if there is significant dependence between term occurrences across the fields. This means that the relevance structure vectors could become very noisy. ICFW was developed for exactly this purpose; to saturate term frequency across fields in order to model the dependence of term occurrence across fields. Therefore, it should be much better in estimating the r_i values in Definition 6.1.

Coming back to the example in Table 6.1, the relevance structure describes the context in which the user wants to rank the list of potential entities. If the seed entity is Emma Watson and the query is *wizards magic fantasy* the context is likely to be actors in movies that are similar to Harry Potter in some ways. More concretely, the relevance structure in this instance would be a vector of values over the document fields. Table 6.2 seeks to clarify this point.

seed entity	r_{plot}	r_{actor}	$r_{\text{character}}$
SE_1	a	b	c
SE_2	a	d	e

Table 6.2: Relevance structure vectors for seed entities from table 6.1. $b > c$ and $d < e$. SE_1 = Emma Watson, SE_2 = Hermione $\text{rs}_1 = [a, b, c]$ and $\text{rs}_2 = [a, d, e]$

This section has not discussed the choice of relevant documents in depth, even though they are needed to define relevance structure in Definition 6.1. This is because in the example we knew beforehand which fields the entities would occur in and that the relevant document would be the Harry Potter movies. Furthermore, so far each information need has been described by a single relevance structure, which is also not necessarily valid. This is where the proposed system comes in. The entire process starts with the user formulating queries with corresponding seed entities and finding relevant documents, which can then be used to define the relevance structures the user is interested in, which in turn can be used to rank the list of potential entities.

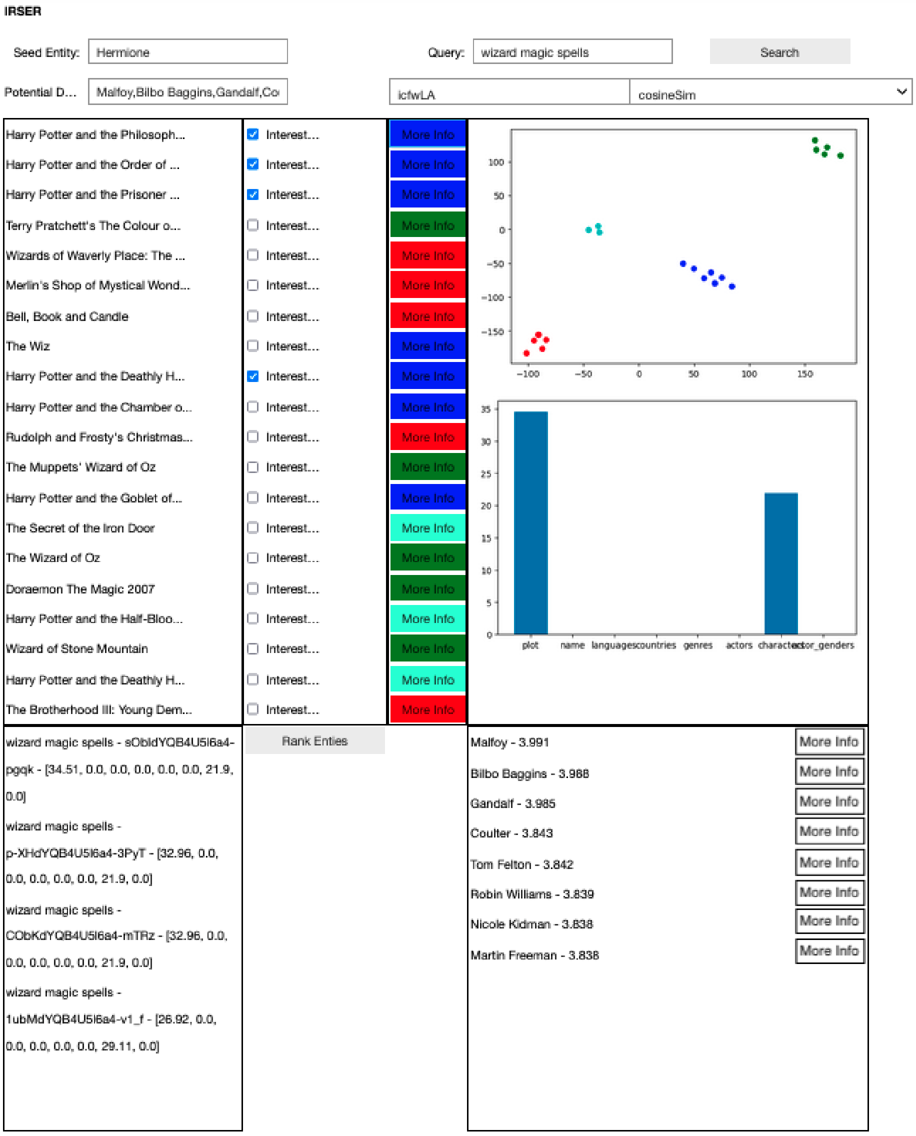
6.5 System Specification

Before diving into the technical description of the system described in this section, it is worth clarifying the notation used:

- SE be a seed entity, i.e. someone we know matches the information need well and can be found in the data
- EoIs = $(\text{pe}_1 \dots \text{pe}_m)$ a list of potential entities of interest who might match the information need.
- $Q(e) = (q_1(e) \dots q_m(e))$ a set of queries. Each query q contains an entity e. This can be either the seed entity or one of the potential entities. For SE = Hermione in Table 6.1 the q(Hermione) = "wizards magic fantasy Hermione".
- $\text{SIM}[\text{rs}(q_i, d_i), \text{rs}(q_j, d_j), \gamma]$ is a function that returns the similarity of two relevance structures. γ denotes a chosen similarity model.
- $\text{ranking}(q, c, M)$ a function that returns a set of relevance structures

Figure 6.2 shows the RSER user interface (UI), which will be used to describe how the system works in detail. The example is based on a simple scenario where the data collection consists of information about movies, actors and characters.

The reason for the simple scenario is to make it easier to follow, but there is no reason why the scenario could not involve more interesting and complex data. Altogether there are six steps in the process. The following details each of these steps.



IRSER

Seed Entity: Hermione

Query: wizard magic spells

Potential D... Malfoy,Bilbo Baggins,Gandalf,Co

icfwLA | cosineSim

Harry Potter and the Philosoph...	☑ Interest...	More Info
Harry Potter and the Order of ...	☑ Interest...	More Info
Harry Potter and the Prisoner ...	☑ Interest...	More Info
Terry Pratchett's The Colour o...	☐ Interest...	More Info
Wizards of Waverly Place: The ...	☐ Interest...	More Info
Merlin's Shop of Mystical Wond...	☐ Interest...	More Info
Bell, Book and Candle	☐ Interest...	More Info
The Wiz	☐ Interest...	More Info
Harry Potter and the Deathly H...	☑ Interest...	More Info
Harry Potter and the Chamber o...	☐ Interest...	More Info
Rudolph and Frosty's Christmas...	☐ Interest...	More Info
The Muppets' Wizard of Oz	☐ Interest...	More Info
Harry Potter and the Goblet of...	☐ Interest...	More Info
The Secret of the Iron Door	☐ Interest...	More Info
The Wizard of Oz	☐ Interest...	More Info
Doraemon The Magic 2007	☐ Interest...	More Info
Harry Potter and the Half-Bloo...	☐ Interest...	More Info
Wizard of Stone Mountain	☐ Interest...	More Info
Harry Potter and the Deathly H...	☐ Interest...	More Info
The Brotherhood III: Young Dem...	☐ Interest...	More Info

wizard magic spells - sObldYQB4U5I6a4-pgqk - [34.51, 0.0, 0.0, 0.0, 0.0, 0.0, 21.9, 0.0]

wizard magic spells - p-XHdYQB4U5I6a4-3PyT - [32.96, 0.0, 0.0, 0.0, 0.0, 0.0, 21.9, 0.0]

wizard magic spells - CObKdYQB4U5I6a4-mTRz - [32.96, 0.0, 0.0, 0.0, 0.0, 0.0, 21.9, 0.0]

wizard magic spells - 1ubMdYQB4U5I6a4-v1_f - [26.92, 0.0, 0.0, 0.0, 0.0, 0.0, 29.11, 0.0]

Rank Enties

Malfoy - 3.991	More Info
Bilbo Baggins - 3.988	More Info
Gandalf - 3.985	More Info
Coulter - 3.843	More Info
Tom Felton - 3.842	More Info
Robin Williams - 3.839	More Info
Nicole Kidman - 3.838	More Info
Martin Freeman - 3.838	More Info

Figure 6.2: User interface for RSER

Step 1: Defining the Entities of Interest

As with any search task, the investigatory process begins with the user having an information need. In the proposed system, the user defines a list of potential entities (EoIs) they are interested in and believe could be found in the data. Furthermore, the user defines a seed entity (SE) that they know is found in the data and corresponds to their information need.

In the example from Table 6.1 and Figure 6.2 the list of potential entities is a mix of character and actor names from movies with magic and wizards: EoIs = [Malfoy, Bilbo Baggins, Alladin, Coulter, Tom Felton, Nicole Kidman, Martin Freeman, Robin Williams]. For Figure 6.2 and the explanation of the process the seed entity SE = Hermione. This suggests that the user is interested in characters found in magical movies, rather than actors.

Step 2: Formulating a Set of Queries

This step consists of the user formulating a set of queries Q that define the context in which they wish to rank the list of potential entities. In our movie example, this context is whether an entity is a character in a movie about magic and wizards. To define this context — with the help of the seed entity — the user should formulate queries that return documents proving that Hermione is a character in such a movie.

These queries should not be too specific, as something like "Harry Potter Hermione" would return the Harry Potter movies, but the associated relevance structures would carry little information in the context of most of the potential entities. Something like "wizards magic spells Hermione" would give better results. The user can also choose the retrieval model and the similarity model. The different possible models are discussed in the Implementation and Evaluation Section (Section 6.7)

Step 3: Choosing Documents of Interest

On the left of Figure 6.2 we can see the produced ranking. To the left of it are some graphs that help navigate the ranking. The top one shows the document on the left clustered based on their relevance structures. The clustering is performed based on the relevance structure of the documents, i.e. the field weights. TSNE clustering is used with the number of clusters calculated using Silhouette Coefficients.

The user can easily examine the relevance structure of each document by clicking the "More info" button which updates the bottom graph. From this graph, the user can get an understanding of exactly how the document is relevant to the query in terms of its structure. In this case, we can clearly see that the document is relevant because of the plot and actor fields.

Using the cluster colours and the bottom graphs the user can easily navigate the results based on their relevance structures. In Figure 6.2 this is clear from the Harry Potter movies all being blue. The user can easily select interesting documents which then appear in the list at the bottom left of the UI. They can re-run the query and add more items to the list. Once they believe they have enough interesting documents the user simply clicks "Rank Entities", which starts the back-end analysis of the potential entities.

Step 4: Calculating Similarity Scores for EoIs

This step is performed by the system, not the user. For each entity in the list of EoIs, we define a set of queries Q(entity) and for each of these queries, we run a search. So for example, for the entity "Malfoy" we would run the query "wizards magic spells Malfoy", just as we ran 'wizards magic spells Hermione" for the seed entity. We would then look at the ranking produced and see if there are relevance structures similar to those defined in the previous step, i.e. the interesting documents were chosen. The similarity of each of the document relevance structures in the ranking for "wizards magic spells Malfoy" would be compared to those chosen by the user for the seed entity, using the similarity model $\text{SIM}[\text{rs}(q_i, d_i), \text{rs}(q_j, d_j), \gamma]$ where d_i is a document of interest chosen by the user, and d_j is a document in the ranking corresponding to the query "wizards magic spells Malfoy". To calculate the similarity for a given EoI, we consider k most similar documents from the EoIs rankings, compared to the documents of interest. Section 6.7 will discuss different options for measuring this similarity and the performance for different values of k.

Step 5: Rank the Potential Entities According to their Relevance to the Information Need

Create the final ranking and allow the user to investigate the underlying queries, documents and fields which have produced said ranking. This is done by sorting the entities based on the SIM scores that were calculated in the previous step.

Step 6: Reasoning

If the user wishes to learn about the system's reasoning for why an EoI gets a certain similarity score they can click the "More Information" button to the right of each EoI. This will show the ranking results for that specific EoIs queries in the main ranking table, as well as the documents of interest that are the reason for its similarity score. By deleting documents of interest that are not relevant to their information needs, the investigator can then "reason" with the system, as the system changes the ranking of the EoIs according to the choices of the user. Effectively the user can overwrite what the system flagged as an interesting document for any of the EoIs, which then gets reflected in the final ranking. The next section will discuss in more detail the nature of the reasoning that the system facilitates.

6.5.1 Algorithmic Description of the System

Algorithm 1 describes the above process more formally

Algorithm 1 Proposed Retrieval Algorithm

```
PersSim = []; # similarity of each potential entity
SE = Hermione; # Seed entity
$\mathrm{RS}_{\mathrm{SE}}$ = RS(SE); # Set of Relevance Structures
$Q$; # set of queries for finding Harry Potter movies
for pe ∈ EoIs do # loop over potentially interesting entities
    RsSim = [] # relevance structure similarities for pe
    for $q$(pe) ∈ $Q(pe)$ do
        $\mathrm{RS}_{\mathrm{pe}}$ = RS(pe) = ranking($q$(pe), $c$, $M$) # fetch RS
        sim = SIM[RS(SE), RS(pe)]
        AddItem(RsSim, topK(sim)) # only consider topK similar RSs
    end for
    AddItem(PersSim, (AVG(RsSim), pe))
end for
Sort(PersSim) # Sort EoIs according to RS similarity with SE
```

6.6 RSER and InvIR

Before discussing the implementation and evaluation of the proposed system, it is worth discussing what exactly makes RSER an investigative search engine, rather than just a search engine, or an exploratory search engine.

To recap, the defining characteristics essential for InvIR are:

1. Complex information needs.
2. Query reformulation.
3. Session-based.
4. Complex results.
5. Complex Data.
6. Transparency.
7. Reasoning.

Complex information needs, query reformulation and the session-based nature of RSER are evident from the previous section. The system considers multiple queries, each of which is comprised of two parts (base query + entity) within a session where the end result is to rank entities of interest. These factors reflect the complexity of the users' information need: They are looking to rank a list of entities, which can be seen as an aspect of their information need, according to other aspects of their information need, described by their various queries. The results are presented in a complex manner, where the relevance structures are explorable both one by one and from clusters. To the best of the author's knowledge, this is the first system that visualises the relationship between relevance and document structures (relevance structures) in this manner. It makes the search much easier as effectively documents with similar relevance structures, i.e. documents that are relevant in the same context have the same colour in the ranking. The complexity of data is also evident, as the search engine is specifically designed to deal with structured data.

Moving onto the aspects of RSER that make it an investigative search engine, rather than an exploratory one. The emphasis on transparency does not have so much to do with the engine design itself, but rather with the underlying retrieval and similarity models. If either of these was replaced by a black-box algorithm a large degree of the transparency of the system would be lost, which would make it an exploratory search engine instead. A degree of transparency is also provided by the way in which the results are presented, as the user has a better idea of which parts of the document contribute to its relevance. In the context of the RSER system, the transparency of the system described above is what facilitates the reasoning aspect of the system. As discussed in Section 2.2, the focus of this thesis with respect to the special aspects of InvIR is on transparency, rather than reasoning. However, with the RSER system, we aim to demonstrate how this transparency can be used to facilitate the understanding of the underlying reasoning within the system and to interact with the system to guide that reasoning.

What it all boils down to is that even if the underlying algorithms are transparent, a user with little knowledge about IR algorithms cannot fully trust the system. That is why the ability of the system to communicate the reasoning of the final ranking of the EoIs to the user and the users' ability to reason together with the system is an important aspect of what makes RSER an investigative search engine. This is why it is important that the user can get a deeper understanding of why each of the EoIs has been ranked high or low. The system accomplishes this by allowing the user to see the rankings that each of the queries for each potential entity has produced and the documents of interest the system has chosen for those query-entity pairs. If the documents of interest are not correct, the user should be able to change them, effectively reasoning together with the system to change the final ranking.

Analysing the existing body of research around reasoning as a cooperative and interactive task between a human and a computer in-depth is out of the scope of this thesis. For example, there exist whole fields of study around concepts such as semantic web, semantic reasoning and reasoning based on knowledge bases that relate to this chapter but are too wide to capture in a clear manner [120, 121, 122, 123]. One of the underlying reasons for this is the fact that due to the technical chapters of the thesis, more focus has been given to transparency. Instead of an in-depth analysis and evaluation of various kinds of reasoning and fields concerned with it, here the aim is to describe what is new about the reasoning that the RSER system facilitates. Systems such as the semantic web use the semantics that information is labelled with to reason for the best possible outcome, the RSER system reasons in a similar manner using the document structures directly. Furthermore, the system communicates its reasoning to the user who can change the underlying logic through which the system has produced the EoI ranking. This means that reasoning becomes a cooperative process between the system and the user, providing an additional layer of transparency.

In summary, what makes the reasoning within RSER special is its focus on the raw document structures, rather than the semantic aspects of documents and the interactive nature of the reasoning process that it facilitates.

Together the seven points described above are what make the RSER system and InvIR system, rather than an IR system of an exploratory search system.

6.7 Evaluation and Analysis

The purpose of this section is to demonstrate the general effectiveness of the RSER system and more importantly, the importance of using ICFW-based field weights when inferring context. As mentioned previously, the experimentation is narrow in its scope and is not aiming to unequivocally show that RSER can be used for complex InvIR scenarios, but instead to demonstrate that it has the potential to do so and that ICFW plays a key part in this. After discussing the implementation of the system and introducing the data collections the following research questions (RQs) will be answered.

RQ1: Is There Value in Using ICFW-based Field Weights to Define Relevance Structures?

RQ2: Which similarity model is the best one?

RQ3: Overall, what does the performance of RSER on the test collection tell us about its effectiveness in general?

6.7.1 Implementation

The proposed RSER system has been implemented using ElasticSearch, python and Jupyter notebooks. Elasticsearch is used to store the data and to perform the initial field-based queries using BM25 and BM25F. For each field in the data collection, we retrieve the top 1000 documents with BM25. Furthermore, we retrieve the top 1000 documents with the BM25F using all the fields and calculate their field-based scores BM25 scores. Hyperparameters are set as $b = 0.8$ and $k_1 = 1.6$, as was the case in Chapter 5. A python library re-ranks the retrieved documents and calculates their field weights using ICFW. We test the system using all three proposed ICFW versions from the previous chapter.

A Jupyter notebook is used to create the user interface (UI) presented in Figure 6.2. The UI is not a part of the evaluation. The system is only evaluated on its performance in terms of a benchmark test collection created specifically for this chapter.

6.7.2 Test Collections

As discussed in the Introduction and Chapter 2, the data structures that IDJ and InvR deals with are highly varied, often containing document types such as emails, legal agreements, spreadsheets, message chains etc. In an ideal scenario, the proposed system would be evaluated on a test collection that has been used in large-scale investigations such as the Panama Papers, or Snowden files. However, the raw data for this kind of information is not openly available. Furthermore, as the retrieval task for the proposed search system tackles is non-standard — as discussed in Section 6.2 — the ground truth, i.e. the optimal ranking of EoIs, has to be defined by us. In order to do this we must possess enough knowledge of the area in question to know what is a good ranking of the entities, meaning the information has to be from an area that the author is familiar with, or even better an area that most readers will be familiar with. This is why we consider movie and Wikipedia data, rather than more complex topics covered by previous investigations. There is a well-known benchmark data collection that relates to InvIR; the Enron email data set. The following details the dataset and explains why we cannot use it.

The Enron email data collection is a vast collection of emails and other electronic communications from the Enron Corporation, a company that was involved in one of the biggest corporate scandals in American history [124]. The collection consists of over 500k emails and other documents that were collected during the investigation of the company's fraudulent accounting practices. The Enron email data collection has been widely studied and analyzed by journalists and others interested in understanding the scandal and its aftermath. Furthermore, it has been used by academics as a benchmark collection for various tasks such as classification, message threading, network analysis, topic modelling etc. [125, 126, 127, 128, 129]. At first glance, it would seem like an ideal test collection for the evaluation here. However, there are three reasons why it is not suitable:

1. None of the existing benchmark versions of the Enron data give a ground truth that fits the InvR task described in this chapter.

2. The author does not possess enough domain knowledge to define base queries, seed entities, potential EoIs or the correct final rankings for the Enron email data.

3. The structure of the data (sender, receiver, subject, body, date) does not have the complexity required to infer "context" to the extent that our system requires.

In summary, since the purpose of this evaluation is not to show that the proposed system "solves" InvIR, but to demonstrate that ICFW is useful in an InvIR scenario both in terms of general performance and visualization, there is no need to use data collections directly related to existing investigations. For this reason, we have chosen datasets that non-expert readers are familiar with, thus making the evaluation more transparent. The underlying datasets used are DBpedia and IMDB.

Example Topic to be Searched

Due to the complexity of the retrieval task, the topics formulated for testing the performance of the system are more complex than in traditional (ad-hoc) IR test collections, such as those seen in Chapters 3, 4 and 5. Listing 1 shows the structure of a test topic.

For each topic, we have the base query ("United States of America female crime thriller"), a seed entity (Uma Thurman), a list of interesting documents that a user would have checked (Kill Bill etc.) and a list of potential entities. For each potential entity, we have defined whether they match the information need, which in this instance corresponds to female actors in crime thrillers from the US. Relevance is judged at 2 levels, 1 = relevant (female actors in movies that clearly fit the crime thriller genre and take place in the US), and 0 (not relevant). Altogether there are 15 topics for IMDB and 10 for DBpedia. One of the reasons for using movie data is that it is easy to semi-automatically create the training set. By filtering the data based on the various fields it is easy to come up with a list of female actors in crime thriller films for example. This, of course, means

Listing 1 Example of a Topic for IMDB

```
{
  "query_id": "3",
  "query": "United States of America female crime thriller",
  "seed_entity": "Uma Thurman",
  "interesting_documents": [
    "UObKdYQB4U5l6a4-MCtu",
    "I-bLdYQB4U5l6a4-YEQo",
    "QebIdYQB4U5l6a4-0A4P",
    "X-bLdYQB4U5l6a4-DD1U",
    "buXGdYQB4U5l6a4-ueHa"
  ],
  "potential_entities": [
    "Lorraine Bracco",
    "Diane Keaton",
    "Jodie Foster",
    "Marlon Brando",
    "Robert Deniro",
    "Ray Liotta",
    "Joe Pesci"],
   "qrels": [
    [
      "3",
      "",
      "Lorraine Bracco",
      "2"
    ],
    ...
    [
      "3",
      "",
      "Robert Deniro",
      "0"
    ]]}
```

that for such an example our system is obsolete if the user knows the data well enough to use filters. However, as discussed before, in an investigative situation the user would most likely not have this kind of knowledge. Furthermore, their information needs are likely to be too complex to be simply communicated with filters. For the RSER system, this would not be a problem and thus the less complex movie-based data is simply used to demonstrate its value. For DBpedia filters could not be used to define relevance, meaning the task had to be done manually, which is also why there are fewer topics.

DBpedia Dataset

The DBpedia data-collection collection from Chapters 3, 4 and 5 is used for the evaluation here as well. The system is evaluated on 10 topics that are summarized in Table 6.3.

Information Need	Base Query	Seed
US presidents that went to Harvard University	United States President Harvard University	Barack Obama
United Nations general secretaries from Africa	united nations secretary general african	Kofi Annan
Books about the second world war in asia	second world war book pacific asia	Guadalcanal Diary
Authors of books about WW1	author book world war one	Barbara Tuchman
Countries with off shore oil rigs	countries with sea based oil reserves drilling platform	Norway
Countries in the Americas with oil	country america oil	Venezuela
Books about the Spanish civil war	book spanish civil war	For Whom the Bell Tolls
Organized crime figures in chicago	organized crime figure chicago mafia	Al Capone
Authors of books about the italian mafia	author book italian mafia	Mario Puzo
Mafia members involved in helping us military in WW2	mafia ww2 world war 2 war effort help new york docks	Luciano

Table 6.3: Information needs, base queries and seed entities for DBpedia test collection

IMDB Dataset

The underlying data for the evaluation is an IMDB database collected by [113]. The data consists of movie, actor and character data. It has been cleaned and is stored in ElasticSearch instance and has the following fields: movie_id, plot, movie_name, movie_languages, movie_countries, movie_genres, actors, characters, actor_genders. Table 6.4 shows the information needs, base queries and seed entities for the IMDB collection.

6.7.3 System Settings

There are four features in the introduced system which can be changed:

1. The underlying retrieval model, M in Algorithm 1.
2. How the relevance structure is defined, i.e. how r_i is defined in Definition 6.1.

3. The similarity metric used to compare the relevance structure, SIM in Algorithm 1.
4. The k value defines how many of the relevance structures are retrieved for each query for each potential entity considered.

Retrieval Model: For the retrieval model we consider FSA-BM25, ICFW-G-BM25, ICFW-GA-BM25, ICFW-LA-BM25, which have been introduced previously in this thesis.

Relevance: Three ways of defining r_i are considered.

1. Field-based BM25 retrieval scores.
2. Field weights assigned by the ICFW model.
3. A product of the two.

Similarity: For the similarity metric γ we consider manhattan distance, cosine distance and a combined metric where the two are multiplied.

k-cutoff: Finally, we try different values of k between 1 and 10. Figure 6.3 shows system performance for different values of k. The analysis assumes $k = 4$ as here we observe good performance for both data collections.

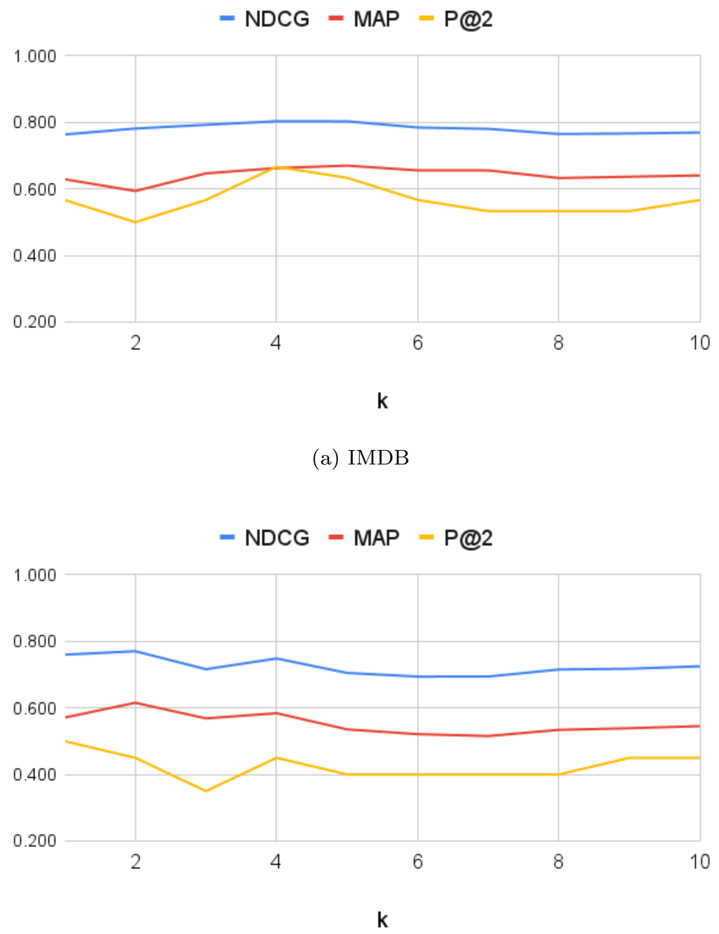
6.7.4 General Performance

Table 6.5 shows the performance of RSER for different underlying retrieval models and similarity metrics at k = 4.

From Table 6.5 and Figure 6.3 it is clear that there is significant variation in the performance of the system depending on how the underlying features are defined. For this reason, it is important that the user interface offers options on these features for the user, as is done in Figure 6.2.

6.7.5 Is There Value in Using ICFW-based Field Weights to Define Relevance Structures?

As discussed earlier, ICFW was the stronger candidate compared to FSA-BM25 as the field weights are less noisy. For FSA, field-based BM25 scores were used to estimate the relevance structure ($r_{i,\text{FSA}} = \text{RSV}_{\text{BM25}}(q, f_i, c)$ in Definition 6.1), whereas for the ICFW-based models the ICFW field weights were used ($r_{i,\text{ICFW}} = w_{\text{ICFW}}(q, f_i, c)$ in Definition 6.1). We also experimented with a combination of ICFW-field weights and the BM25 score, where the two were multiplied to estimate $r_{i,\text{ICFW-BM25}}$: ($r_i = \text{RSV}_{\text{BM25}}(q, f_i, c) \times w_{\text{ICFW}}(q, f_i, c)$ in Definition 6.1)



(a) IMDB

(b) DBpedia

Figure 6.3: System performance for different values of k. k equals the different cut-off points explained in Section 6.7.3, i.e. how many of the relevance structures are retrieved for each query for each potential entity considered.

Table 6.5 demonstrates the feasibility of using ICFW-based field weights for defining the relevance structures in RSER. The results are relatively noisy due to the small number of evaluation topics per data collection (especially the P@2 column). However, we can observe that there are no instances where the FSA-BM25-based model outperforms the ICFW-based models. It is worth noting that there are important differences between the two data collections: In general, the benefits of using ICFW are greater for the IMDB dataset. A likely reason for this is that the document structure for IMDB is much more complex than it is for DBpedia, with 9 fields for the former and 5 for the latter. Furthermore, the fields are much more diverse for IMDB. For example, the title of a Wikipedia page relates closely to its body, similar title and related titles, whereas for IMDB — apart from movie plot and title — the document fields are much more different semantically.

As a general trend, we can say that ICFW-LA has the most robust performance across similarity metrics and datasets. This makes sense intuitively as ICFW considers the field level term metrics when calculating the lambda scaling parameter, whereas ICFW-G and ICFW-GA average over the fields (Definitions 5.15, 5.16, 5.17).

6.7.6 Which Similarity Model is the Best One?

The experimentation does not show that one similarity model is better than the others. The results suggest that whether cosine similarity or manhattan distance is better, depends heavily on the data collection. For DBpedia, manhattan distance would seem to produce better results when ICFW models are used and worse results if a combination of ICFW and BM25 is used. Overall the best results are obtained by using ICFW-LA together with manhattan distance. A possible reason for this is that due to the simpler document structure, the model needs to consider the degree of relevance for each field, as well the relative importance of each field. For the IMDB dataset, cosine similarity does better than manhattan similarity. This is likely to be because the relative importance of fields is more important than their degree.

To clarify this point, consider two queries with two seed entities, one for each data collection: "United States President Harvard University" with the seed entity "Barack Obama" for DBpedia and "Actors in Italian mafia movies" with the seed entity "Al Pacino" for the IMDB collection. For DBpedia important documents would include Wikipedia articles such as "Barack Obama's timeline" for IMDB movies such as "The Godfather".

For IMDB we would like to rank high entities where the query terms "Italian mafia" occur in the plot and/or description fields and the entity name (Al

Pacino), occurs in the actor_names field. If the entity name occurs in any other field, the context is automatically wrong, as we are only interested in actors, not characters for example. What this means is that the relative importance of fields in terms of the relevance structure is of large importance, most likely more so than the actual degree of relevance for any individual field. For DBpedia things are not as straightforward. Since there are not as many fields, the system will find it more difficult to differentiate context based on whether a field is relevant or not, instead the degree of relevance will need to be considered. For example, the Wikipedia timeline article should be about the EoI in question, but there is no field that lists the important entities in a document for example. So the system would need to distinguish the occurrence of the entity name (Obama) from other query terms, such as president.

To summarise, with fewer fields the term-level occurrences of query terms become more important that field-level occurrences, which is likely to be the reason why for DBpedia manhattan similarity does better than cosine similarity and why for IMDB the opposite is true.

6.7.7 Overall, what does the performance of RSER on the test collection tell us about its effectiveness in general?

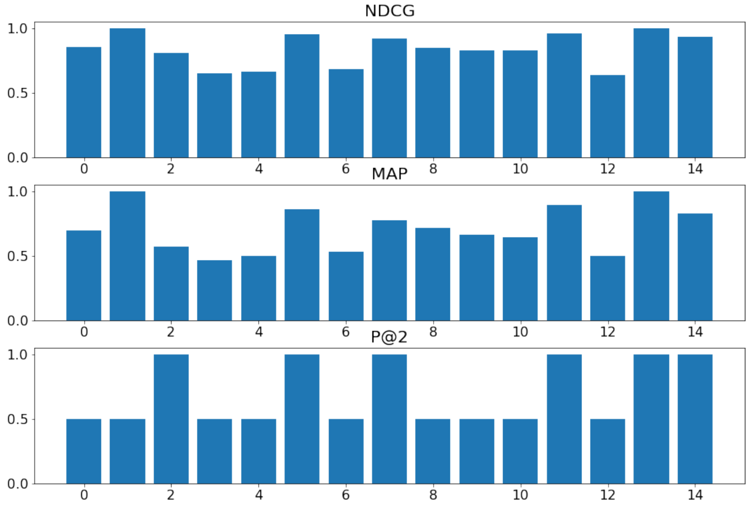


Figure 6.4: Query-based Accuracies of RSER with ICFW-LA and Manhattan Similarity on DBpedia

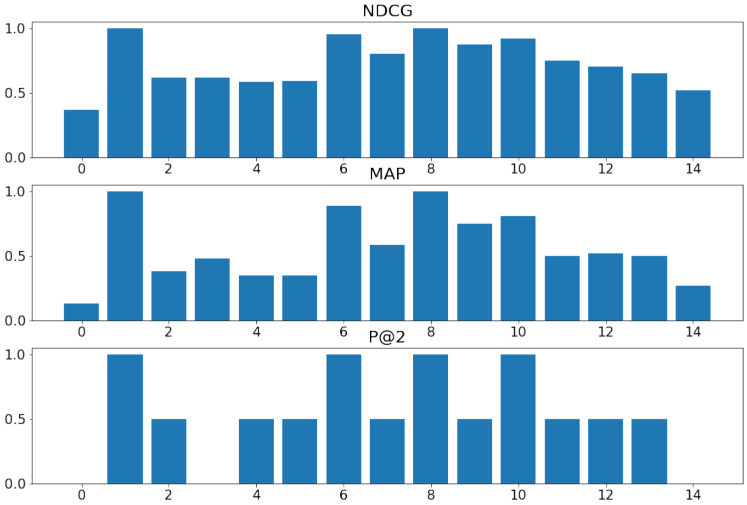


Figure 6.5: Query-based Accuracies of RSER with ICFW-LA and Cosine Similarity in IMDB

Figures 6.5 and 6.4 show the query level accuracies for RSER with DBpedia and IMDB respectively. We can see that for all three accuracy metrics, the performance of RSER is relatively steady across queries. For only three of the queries does MAP drop below 0.45 and for NDCG below 0.75. Precision@2 is 0.5, or 1 for all queries, meaning the first two entities in the final ranking usually provide at least one good true positive entity. This suggests that RSER is indeed accurate enough to help investigators rank entities in terms of their context relative to a seed entity, albeit for two narrow test collections.

6.7.8 Discussion

The experimentation has demonstrated that the entity ranking aspect of RSER is able to rank movie and Wikipedia-based entities based on their context relative to seed entities relatively well. A user-based evaluation with investigation-related data would be required to unequivocally show that the system can be used in investigations. Such a study is outside of the scope of this thesis, which spent the first three content chapters developing the technical solutions required for the implementation of RSER. However, the experimentation does suggest that the approach is valid and warrants further study.

6.8 Summary, Conclusions and Contributions

This chapter covered the following issues:

- The introduction of an InvIR task where a user ranks a list of potentially interesting entities based on their information need.
- A proposed method for tackling the above retrieval task, denoted **I**nvestigatory Search for **R**elevance **S**tructure-Based **E**ntity **R**anking (RSER).
- Introducing the concept of relevance structures and demonstrating how they can be used to further knowledge gained from the data by the user.
- The system was evaluated on a test collection built for the task specifically.

The main conclusions were:

- RSER ranks entities of interest well enough to suggest that such a system could be useful in an InvIR scenario.
- The system works much better with ICFW than any other SDR methods, to the extent that the system would not be feasible without ICFW

The main contributions are:

- An investigative search engine prototype denoted RSER.
- A formal evaluation of the search engine demonstrating how relevance structures can be used in InvIR scenarios.

Information Need	Base Query	Seed
Actors that have appeared in westerns with Clint Eastwood	Clint Eastwood Western	Wallach
character names in italian mafia movies	italian mafia	Vito Corleone
actors names in italian mafia movies	italian mafia	Pacino
Female lead characters in movies about crime in the united states of america	United States of America female crime thriller	Uma Thurman
Movies with Harrison Ford that take place in United States with action	Harrison Ford United States action	Fugitive
characters in comedies with Jim Carrey about Christmas	comedy Jim Carrey Christmas	Grinch
actors in films with wizards and magic	wizard magic spells	Daniel Radcliffe
actors in films with wizards and magic	wizard magic spells	Hermione
characters in movies with Bruce Willis about Christmas	Christmas Bruce Willis	John McClane
movies with Bruce Willis and Samuel Jackson	Bruce Willis Samuel Jackson	Die Hard 3
marx brothers	black and white commedy	Harpo Marx
male actors in action films with Schwarzenegger	action film arnold Schwarzenegger male actors	Dolph Lundgren
male characters in action films with Schwarzenegger	action film arnold Schwarzenegger male characters	John Matrix
german speaking movies set in Berlin during war times	Berlin German Language War	Good by Lenin
Characters that have appeared in westerns with Clint Eastwood	Clint Eastwood Western	Tuco

Table 6.4: Information needs, base queries and seed entities for IMDB test collection

weights	similarity	DBpedia			IMDB		
		ndcg	p2	map	ndcg	p2	map
FSA-BM25	cosine	0.765	0.600	0.557	0.750	0.560	0.567
	manhattan	0.758	0.500	0.550	0.716	0.567	0.540
	cos*man	0.762	0.650	0.562	0.674	0.600	0.443
ICFW-G	cosine	0.715	0.600	0.492	0.783	0.567	0.621
	manhattan	0.785	0.550	0.619	0.736	0.533	0.559
	cos*man	0.739	0.450	0.558	0.704	0.567	0.496
ICFW-GA	cosine	0.792	0.550	0.642	0.770	0.567	0.622
	manhattan	0.818	0.600	0.679	0.713	0.500	0.534
	cos*man	0.752	0.550	0.612	0.683	0.467	0.464
ICFW-LA	cosine	0.749	0.450	0.584	**0.803**	0.633	**0.670**
	manhattan	**0.823**	**0.700**	**0.691**	0.731	0.533	0.567
	cos*man	0.763	0.650	0.603	0.675	0.500	0.469
ICFW-G x BM25	cosine	0.724	0.450	0.561	0.660	0.589	0.415
	manhattan	0.729	0.500	0.558	0.689	0.533	0.497
	cos*man	0.775	0.600	0.592	0.689	**0.733**	0.405
ICFW-GA x BM25	cosine	0.703	0.450	0.510	0.662	0.700	0.381
	manhattan	0.701	0.450	0.533	0.663	0.333	0.460
	cos*man	0.681	0.400	0.481	0.662	0.700	0.381
ICFW-LA x BM25	cosine	0.733	0.600	0.554	0.656	0.633	0.429
	manhattan	0.750	0.500	0.595	0.708	0.533	0.522
	cos*man	0.731	0.400	0.535	0.643	0.700	0.376

Table 6.5: Experimentation results for RSER

Chapter 7

Conclusions

The initial research objective for this thesis was to develop tools, approaches and models that would help with data-driven investigations. To this end, the thesis began by defining investigative retrieval as a sub-task of exploratory search with an emphasis on transparency and reasoning. By doing so, the high-level initial question was framed within the specific research area of IR. The focus on transparency — combined with the inherent complexity of data collections that investigators deal with — guided the choice of technical contributions in this thesis. More specifically, the methods developed in this thesis have contributed to the wider field of IR, by proposing reliable standards for analytical SDR. The thesis finished with a chapter where these technical contributions are applied to InvIR.

Probabilistic models, such as the BM25 and LM have become the standard for non-structured (atomic) retrieval, especially if the use of learn-to-rank models is not warranted. No such widely accepted standard exists for structured document retrieval (SDR). The fielded extension of the BM25 — the BM25F — could be considered the best candidate. However, without optimization, it does not benefit from the document structure. The main technical contribution of this thesis has been the introduction of information content field weighting (ICFW); a new field weighting method for SDR. ICFW is analytical, works without optimization, but can benefit from it and leverages the structure of the documents effectively. These three characteristics are what make the model a potential candidate for a new standard SDR model and what make its use in InvIR feasible.

ICFW and related concepts were introduced throughout the thesis. There are four steps, corresponding to chapters, which covered various aspects of the model:

- **Initial model description:** Theoretical and intuitive justification for the use of information content in field weighting and initial study (Chapter 3).

- **Formal constraints for SDR:** In order to better understand where information content-based field weighting does well and where it does not, constraints for SDR were formalized, allowing for analytical evaluation and comparison of SDR models (Chapter 4).

- **Cross-field term frequency saturation in ICFW:** From the previous step, cross-field term frequency was identified as the main property missing from the initial study model. This step analyses and identifies the best ways to apply this saturation in information content-based field weighting, presenting the ICFW method. (Chapter 5).

- **Investigative search application:** The usefulness of ICFW for InvIR was demonstrated with the help of a prototype search engine where entities of interest are ranked according to the context in which they are found in a data collection. (Chapter 6).

Chapter 3 described intuitive and theoretical justifications for the use of information content for field weighting. The intuitive justification is closely related to the original justification for the TF-IDF model by Spark-Jones from more than 50 years ago. In their conceptual model for TF-IDF weighting, the term weight is defined through its exhaustivity and specificity. The core revelation by Spark-Jones' original definition of the IDF was to argue that

> It [specificity] should be interpreted as a statistical rather than semantic property of index terms. [20]

Meaning specificity should be automatically calculated using collection metrics, rather than defined manually.

A central aspect of this thesis is to transfer methods from atomic retrieval to SDR. The most obvious example of this is the SDR constraints, directly inspired by the (atomic) retrieval constraints by Fang et al. [26]. However, a similar transfer underlies the conceptual model and therefore the intuitive justification of ICFW. By defining the conceptual model underlying SDR as one where a field-level retrieval score represents the exhaustivity of a field and the assigned field weights its specificity, this thesis has argued that the specificity (field weight) should be perceived as a statistical property of the field, rather than a semantic one. Another similarity of this thesis to the work of Spark-Jones is proposing that specificity should be calculated as the negative log of the probability of a field, as is the case for the specificity of terms with the IDF.

Two approaches for the theoretical justification of using the information content for field weighting — calculated as the negative log probability of a field

— have been given. One is related to the information-theoretic justification to the IDF by Aizawa [103] and one to the justification for the use of information content in DFR retrieval, relating to research on semantic information theory by Hintikka [39, 40]. Each of these justifications has issues. This is not a huge problem, as it is not the intention of this thesis to provide a complete, formal mathematical justification for ICFW, but to demonstrate that there are intuitive and theoretical justifications that can be used to explain why the model works. Together with the intuitive and theoretical justifications for using information content for field weighting, Chapter 3 also contains a study that uses a simplified version of ICFW and demonstrates how it performs on two test collections. It is obvious from the results that the model does very well on some collections and very badly on others. The reasons behind this led to the next chapter.

Chapter 4 introduced formal constraints for SDR. The motivation came from the findings of Chapter 3 and existing discussion within SDR research. In many ways existing research — especially with respect to the BM25F — seemed to agree that summing together field-based scores was not a robust solution to SDR, as it assumes the occurrences of a term across fields to be independent of each other. This was the main motivation for the BM25F, where weighted term occurrences are first summed together and the retrieval model is then applied over this flattened document representation. The findings from the study in the previous step however suggested that this is not always the case. Models that sum field-level scores — BM25-FIC included — outperformed the BM25F for some test collections. So it would seem that sometimes it is better to consider the structure explicitly, rather than worry about cross-field term frequency dependence. This trade-off had not been examined in depth in the past, which led to the formalization of retrieval constraints for SDR in this thesis. The intuitive definitions for the constraints can be found in Table 4.1 and are repeated below. One of the main findings in Chapter 4 was that none of the existing SDR approaches satisfies all four constraints. Furthermore, there is a trade-off between FSA and TFA-based models and between the TD-Constraint and the FD-Constraint. The motivation for the next step (Chapter 5) came from this trade off: Would it be possible to define the ICFW model in a way that guaranteed the satisfaction of all four constraints?

The first two content chapters (Chapters 3 and 4) together with existing research — especially by Robertson et. al on the BM25F [69, 70] — show that in order for a model to satisfy the TD-Constraint, it would need to saturate term frequency across fields. However, if we wanted the model to satisfy the FD-Constraint, it would also need to consider the field-based score. The latter condition was already satisfied by the BM25-FIC model from Chapter 3, so Chapter 5 focused on the TD-Constraint. In order to saturate term frequency

Constraint	Abbr.	Intuition
Term distinctiveness	TD-Co	Adding unseen query terms to a document should increase the retrieval score more than adding query terms already considered
Field distinctiveness	FD-Co	Adding a query term to a new field should increase the retrieval score more than adding it to a field where it already occurs
Term importance	TI-Co	A model should consider the importance of a term on a field level, rather than document-level
Field importance	FI-Co	A model should be able to boost or decrease the weight given to a field, based on some notion of field importance

Table 7.1: Intuition underlying formal constraints for SDR. Field refers to a field of a document; e.g. *abstract* or *author*. Table repeated from Chapter 4.

across fields, the proposed ICFW model not only considers the information content of a document field at a collection field level but at the document level as well. In simple terms, this meant the addition of another component (ICD) and a scaling feature lambda to the field weight calculation:

$$w_{\mathrm{ICFW}}(f_i, d, F_i) = \mathrm{ICF}(f_i, d, F_i) \tag{7.1}$$

$$\rightarrow w_{\mathrm{ICFW},\vec{\lambda}}(f_i, d, F_i) = \mathrm{ICF}(f_i, d, F_i) + \lambda_i \,\mathrm{ICD}(f_i, d) \tag{7.2}$$

See Chapter 5 for formal definitions. Extensive analysis is performed to show that by analytically setting lambda ICFW satisfies all four constraints, albeit with some conditions regarding query term specificities, collection metrics and the degree of within-field term frequency saturation by the underlying model. Table 5.1 from Chapter 5 (repeated below) demonstrates how the ICFW method differs from existing approaches with respect to the SDR constraints and makes obvious why it is expected to outperform them. Extensive formal evaluation

	Term Distinct. TD-Co	Field Distinct FD-Co	Term Import. TI-Co	Field Import. FI-Co
FSA	NO	Conditional	YES	YES
PRMS	NO	Conditional	NO	YES
BM25F	Conditional	NO	NO	YES
MLM	Conditional	NO	NO	YES
FSDM	Conditional	NO	NO	YES
BM25-FIC	NO	Conditional	YES	YES
ICFW	**Conditional**	**Conditional**	**YES**	**YES**

Table 7.2: Constraint satisfaction of SDR models, including ICFW. Table repeated from Chapter 5.

using established test collection is performed to demonstrate that ICFW does indeed outperform existing analytical approaches for SDR.

The contributions of Chapters 3, 4 and 5 represent the main technical contributions of this thesis from a technical IR perspective. Together they have developed a model that fulfils the three conditions for a potential new standard model for SDR by being analytical, not requiring optimization and leveraging the document structure properly. These aspects are also what make the model viable for use in InvIR.

The last chapter introduces an investigative search engine which uses the ICFW-based field weights to infer the contexts in which entities occur in a data collection and use these contexts to rank entities of interest. The system is called **R**elevance **S**tructure-based **E**ntity **R**anking (RSER). Chapter 6 introduces the system in detail and evaluates its performance. It was demonstrated that the system has the potential to be used in investigative scenarios, that it can help users to understand the structure of the data they are dealing with better, give them insight on how different document-query pairs provide evidence and that ICFW is a core component of the system, without which its performance is significantly worse.

The purpose of this line of enquiry is not "solve" or "automate" InvIR, but to demonstrate how the field weights inferred by ICFW can be used to perform investigative search more effectively compared to the BM25F or another existing approach. This is hugely valuable as it shows that the idea of treating field weights as a statistical, rather than a semantic property of document fields, not only increases retrieval performance but also opens up new opportunities for more transparent and complex discovery. Demonstrating how ICFW weights are essential for the proper functioning RSER brings together the overall contribution of this thesis, which was to develop technical methods (ICFW) that can be used in investigative scenarios to help investigators (RSER).

In conclusion, this thesis has paved the way for establishing new reliable standards for SDR and for developing retrieval methods specific to data-driven investigations. This was done by clearly defining what would be expected of a reliable standard SDR model, through the definition of formal SDR constraints, by formulating a model that adheres to those constraints (ICFW) and by demonstrating the model's value for investigative search.

7.1 Future Perspectives

Many of the methods discussed and proposed in this thesis are concerned with relatively new and untouched areas. Within atomic retrieval, there are almost two decades of active research with respect to retrieval constraints. This thesis and related publications represent the first efforts to do the same for SDR. Furthermore, the notion of considering field weights as statistical properties of fields and adjusting the scale of cross-field term frequency saturation is quite different from current research focuses in IR. And finally, the research area of InvIR discussed in this thesis is highly topical in a world where there is an abundance of data, from which facts need to be inferred in a transparent manner.

Due to the relative novelty and topicality of the areas discussed in this thesis, there are many potential high-level options for continuing the research in the future. These options include, but are not limited to:

- More SDR constraints.
- Different ways of estimating the optimal scale of term frequency saturation across fields.
- New methods for optimising the ICFW model using training data.
- Adapting ICFW for hierarchical and connected data.
- Other search tasks and solutions that use the field weights produced by ICFW.
- New approaches and models directed specifically at InvIR.
- User-based evaluation of RSER and other InvIR tools.

Appendix A

Appendix

A.1 Term Distinctiveness and Scale Parameter Lambda

The underlying idea of the scale threshold theorem is that there exists a threshold for λ, above which the model satisfies the term distinctiveness constraint.

Let $q = \{t_1, \ldots, t_n\}$ be a query, d be a document with $n(t_a, f_i, d)$ occurrences of query term t_a in field f_i and $n(t_b, \overline{f}, d)$ occurrences of query term t_b in an average field $\overline{f}$. Let $\overline{d}$ be an amended version of document d where the occurrences of t_b are replaced with occurrences of t_a.

Theorem A.1. *Given terms t_a and t_b, if $\lambda > \lambda_{TD\text{-}th}$, then $RSV(d) > RSV(\overline{d})$. Regarding term frequencies this means $n(\mathrm{t_a}, f_i, d) = n(\mathrm{t_a}, f_i, \overline{d})$ and $n(\mathrm{t_b}, \overline{f}, d) = n(\mathrm{t_a}, \overline{f}, \overline{d})$*

$$\begin{aligned} \forall (t_a, t_b) \in q : \lambda > \lambda_{TD\text{-}th}(\mathrm{t_a}, \mathrm{t_b}, d, f_i) \\ \Rightarrow RSV_{ICFW,\vec{\lambda}}(q, d, c) > RSV_{ICFW,\vec{\lambda}}(q, \overline{d}, c) \end{aligned} \tag{A.1}$$

Proof. Following Def. 5.8 for $\lambda_{\text{TD-th}}$, the inequality becomes:

$$\lambda > \frac{\log \frac{\mathrm{df}(\mathrm{t_b}, \overline{F}) |\overline{F}|^{\Omega\Psi}}{\mathrm{df}(\mathrm{t_a}, \overline{F})^{\Omega\Psi} |\overline{F}|}}{\log \frac{m^{\Omega+1}\, \mathrm{ff}(\mathrm{t_a}, \overline{d})^{\Omega(\Psi+1)}}{m^{\Omega(\Psi+1)}\, \mathrm{ff}(\mathrm{t_a}, d)^{\Omega+1}}} \tag{A.2}$$

Considering the numerator first:

$$\log \frac{\mathrm{df}(\mathrm{t_b}, \overline{F}) |\overline{F}|^{\Omega\Psi}}{\mathrm{df}(\mathrm{t_a}, \overline{F})^{\Omega\Psi} |\overline{F}|} = \log \frac{\frac{\mathrm{df}(\mathrm{t_b}, \overline{F})}{|\overline{F}|}}{\frac{\mathrm{df}(\mathrm{t_a}, \overline{F})^{\Omega\Psi}}{|\overline{F}|^{\Omega\Psi}}} \tag{A.3}$$

Following Eqn. (5.1) for the definition of probabilities and Eqn. (A.3) we obtain,

$$\log \frac{P(\mathrm{t_b}, \overline{f}|\overline{F})}{P(\mathrm{t_a}, \overline{f}|\overline{F})^{\Omega\Psi}} = \log P(\mathrm{t_b}, \overline{f}|\overline{F}) - \Omega\Psi \log P(\mathrm{t_a}, \overline{f}|\overline{F}) \tag{A.4}$$

Following Definition 5.3 we can re-write Eqn. (A.4) to obtain,

$$\log P(\mathrm{t_b}, \overline{f}|\overline{F}) - \Omega\Psi \log P(\mathrm{t_a}, \overline{f}|\overline{F}) = \Omega\Psi\, \mathrm{ICF}(\overline{f}, \overline{d}) - \mathrm{ICF}(\overline{f}, d) \tag{A.5}$$

Moving onto the denominator,

$$\log \frac{m^{\Omega+1}\, \mathrm{ff}(\mathrm{t_a}, \overline{d})^{\Omega(\Psi+1)}}{m^{\Omega(\Psi+1)}\, \mathrm{ff}(\mathrm{t_a}, d)^{\Omega+1}} = \log \frac{[\frac{\mathrm{ff}(\mathrm{t_a},\overline{d})}{m}]^{\Omega(\Psi+1)}}{[\frac{\mathrm{ff}(\mathrm{t_a},d)}{m}]^{\Omega+1}} \tag{A.6}$$

Inserting Eqn. (5.2) to Eqn. (A.6) and transforming the log expression we obtain,

$$\log \frac{P(f_i|\overline{d})^{\Omega(\Psi+1)}}{P(f_i|d)^{\Omega+1}} = \Omega(\Psi+1) \log P(\mathrm{t_a}, f_i|\overline{d}) - \Omega \log P(\mathrm{t_a}, f_i|d) - \log P(\mathrm{t_a}, \overline{f}|d) \tag{A.7}$$

Following Definition 5.3 we can re-write Eqn. (A.7) to obtain

$$\log \frac{P(f_i|\overline{d})^{\Omega(\Psi+1)}}{P(f_i|d)^{\Omega+1}} = -\Omega(\Psi+1)\, \mathrm{ICD}(f_i, \overline{d}) + \Omega\, \mathrm{ICD}(f_i, d) + \mathrm{ICD}(\overline{f}, d) \tag{A.8}$$

$$\lambda > \frac{\Omega\Psi\, \mathrm{ICF}(\overline{f}, \overline{d}) - \mathrm{ICF}(\overline{f}, d)}{-\Omega(\Psi+1)\, \mathrm{ICD}(f_i, \overline{d}) + \Omega\, \mathrm{ICD}(f_i, d) + \mathrm{ICD}(\overline{f}, d)} \tag{A.9}$$

Inserting Eqn. (A.5) and Eqn. (A.8) to Eqn. (A.2) and solving for Ω we obtain,

$$\Omega < \frac{\mathrm{ICF}(\overline{f}, d) + \lambda\, \mathrm{ICD}(\overline{f}, d)}{\lambda(\Psi+1)\, \mathrm{ICD}(f_i, \overline{d}) - \lambda\, \mathrm{ICD}(f_i, d) + \Psi\, \mathrm{ICF}(\overline{f}, \overline{d})} \tag{A.10}$$

This inequality only holds if

$$-\lambda(\Psi+1)\, \mathrm{ICD}(\overline{f}, \overline{d}) + \lambda\, \mathrm{ICD}(f_i, d) - \Psi\, \mathrm{ICF}(\overline{f}, \overline{d}) > 0 \tag{A.11}$$

$$\Psi < \frac{\lambda\, \mathrm{ICD}(\overline{f}, \overline{d}) + \lambda\, \mathrm{ICD}(f_i, d)}{-\lambda\, \mathrm{ICD}(\overline{f}, \overline{d}) - \mathrm{ICF}(\overline{f}, \overline{d})} \tag{A.12}$$

and

$$-\Omega(\Psi+1)\, \mathrm{ICD}(f_i, \overline{d}) + \Omega\, \mathrm{ICD}(f_i, d) + \mathrm{ICD}(\overline{f}, d) > 0 \tag{A.13}$$

$$\Omega < \frac{-\mathrm{ICD}(\overline{f}, d_i)}{(\Psi+1)\, \mathrm{ICD}(f_i, \overline{d}) + \mathrm{ICD}(f_i, d_i)} \tag{A.14}$$

Expanding the denominator from Equation (A.10) we obtain,

$$\Omega < \frac{\mathrm{ICF}(\overline{f}, d) + \lambda\,\mathrm{ICD}(\overline{f}, d)}{\begin{array}{l}\mathrm{ICF}(f_i, \overline{d}) + \lambda\,\mathrm{ICD}(f_i, \overline{d}) + \Psi\,\mathrm{ICF}(\overline{f}, \overline{d}) \\ \quad + \Psi\lambda\,\mathrm{ICD}(f_i, \overline{d}) - \mathrm{ICF}(f_i, d) - \lambda\,\mathrm{ICD}(f_i, d)\end{array}} \tag{A.15}$$

Following Definition. 5.7, Eqn. (5.5) and Eqn. (5.6) Eqn. (A.15) is re-written:

$$\frac{\mathrm{S}_{\mathrm{contr}}(\mathrm{t_a}, f_i, d)}{\mathrm{S}_{\mathrm{contr}}(\mathrm{t_b}, \overline{f}, d)} < \frac{w_{\mathrm{icfw}}(\overline{f}, d)}{w_{\mathrm{icfw}}(f_i, \overline{d}) + \Psi w_{\mathrm{icfw}}(\overline{f}, \overline{d}) - w_{\mathrm{icfw}}(f_i, d)} \tag{A.16}$$

Rearranging Eqn. (A.16) we obtain,

$$\begin{aligned} &w_{\mathrm{icfw}}(\overline{f}, d)\,\mathrm{S}_{\mathrm{contr}}(\mathrm{t_b}, \overline{f}, d) + w_{\mathrm{icfw}}(f_i, d)\,\mathrm{S}_{\mathrm{contr}}(\mathrm{t_a}, f_i, d) > \\ &\quad w_{\mathrm{icfw}}(f_i, \overline{d})\,\mathrm{S}_{\mathrm{contr}}(\mathrm{t_a}, f_i, \overline{d}) + w_{\mathrm{icfw}}(\overline{f}, \overline{d})\,\mathrm{S}_{\mathrm{contr}}(\mathrm{t_a}, \overline{f}, \overline{d}) \end{aligned} \tag{A.17}$$

Assuming the term frequencies from the theorem, the retrieval score difference is only dependent on the score contributions of term $\mathrm{t_a}$ in field f_i and term $\mathrm{t_b}$ in field $\overline{f}$. For $\overline{d}$ the same is true for the score contributions of term $\mathrm{t_a}$ in field f_i and term $\mathrm{t_a}$ in field $\overline{f}$. Following Definition 5.5 we rewrite Eqn. (A.17) and obtain the implicated inequality from the theorem.

□

A.2 Field Distinctiveness and Scale Parameter Lambda

The underlying idea of the scale threshold theorem is that if there exists a λ that is greater than 0 and greater than a certain threshold ($\lambda_{\text{TD-th}}$), and λ is higher than said threshold the model satisfies the field distinctiveness constraint.

Let $q = \{t_1, \ldots, t_n\}$ be a query, d a document with T occurrences of term t in field f_i and z occurrences of term t in an average field $\overline{f}$. Let $\overline{d}$ be an amended version of d, where the occurrences of term t in $\overline{f}$ have been moved to f_i and z occurrences of non-query terms have removed from f_i and added to $\overline{f}$. These non-query terms ensure that theorem is only concerned with query term occurrences, rather than document lengths.

Theorem A.2. *Given term t and field f_i and $\overline{f}$, if $\lambda > \lambda_{TD\text{-}th}$ and $\lambda > 0$, then $RSV(d) > RSV(\overline{d})$.*

$$\begin{aligned} &\lambda < \lambda_{TD\text{-}th}(t, d, f_i) > 0 \\ &\quad \Rightarrow RSV_{ICFW,\vec{\lambda}}(q, d, c) > RSV_{ICFW,\vec{\lambda}}(q, \overline{d}, c) \end{aligned} \tag{A.18}$$

Proof. Following Definition 5.11 the inequality becomes:

$$\lambda < \frac{\log \frac{[\frac{\mathrm{df}(t,\overline{F})}{|\overline{F}|}]^{\Psi}}{[\frac{\mathrm{df}(t,F_i)}{|F_i|}]^{\zeta-1}}}{\log \frac{[\frac{\mathrm{ff}(t,\overline{d})}{m}]^{\zeta}}{[\frac{\mathrm{ff}(t,d)}{m}]^{1+\Psi}}} \tag{A.19}$$

$$\lambda < \frac{-\log[\frac{\mathrm{df}(t,F_i)}{|F_i|}]^{\zeta-1} + \log[\frac{\mathrm{df}(t,\overline{F})}{|\overline{F}|}]^{\Psi}}{-\log[\frac{\mathrm{ff}(t,d)}{m}]^{1+\Psi} + \log[\frac{\mathrm{ff}(t,\overline{d})}{m}]^{\zeta}} \tag{A.20}$$

$$\lambda < \frac{(\zeta - 1)\,\mathrm{ICF}(f_i, d) - \Psi\,\mathrm{ICF}(\overline{f}, d)}{(1 + \Psi)\,\mathrm{ICD}(f_i, d) - \zeta\,\mathrm{ICD}(f_i, \overline{d})} \tag{A.21}$$

$$\tag{A.22}$$

Conditional on $(1 + \Psi)\,\mathrm{ICD}(f_i, d) - \zeta\,\mathrm{ICD}(f_i, \overline{d}) > 0$ Equation (A.19) can be transformed to obtain

$$\mathrm{ICF}(f_i, d) + \lambda\,\mathrm{ICD}(f_i, d) + \Psi\,\mathrm{ICF}(\overline{f}, d) + \Psi\lambda\,\mathrm{ICD}(\overline{f}, d) > \tag{A.23}$$

$$\zeta\,\mathrm{ICF}(f_i, \overline{d}) + \zeta\lambda\,\mathrm{ICD}(f_i, \overline{d}) \tag{A.24}$$

Following Definition 5.4 we obtain

$$w_{\text{icfw}}(f_i, d) + w_{\text{icfw}}(\overline{f}, d) > \zeta w_{\text{icfw}}(f_i, \overline{d}) \tag{A.25}$$

Inserting Equations (5.11) and (5.19) we obtain

$$\frac{w_{\text{icfw}}(f_i, d) + w_{\text{icfw}}(\overline{f}, d)\,\frac{\mathrm{S}_{\text{contr}}(t, \overline{f}, d)}{\mathrm{S}_{\text{contr}}(t, f_i, d)}}{w_{\text{icfw}}(f_i, \overline{d})} > \frac{\mathrm{S}_{\text{contr}}(t, f_i, \overline{d})}{\mathrm{S}_{\text{contr}}(t, f_i, d)} \tag{A.26}$$

Rearranging Equation (A.26) we obtain:

$$\begin{aligned} & w_{\text{icfw}}(f_i, d)\,\mathrm{S}_{\text{contr}}(t, f_i, d) + w_{\text{icfw}}(\overline{f}, d)\,\mathrm{S}_{\text{contr}}(t, \overline{f}, d) \\ & > w_{\text{icfw}}(f_i, \overline{d})\,\mathrm{S}_{\text{contr}}(t, f_i, \overline{d}) \end{aligned} \tag{A.27}$$

□

Bibliography

[1] T. Ketola and T. Roelleke, "BM25-FIC: Information Content-based Field Weighting for BM25F," in *BIRDS@SIGIR*, 2020.

[2] T. Ketola and T. Roelleke, "Formal Constraints for Structured Document Retrieval," in *SIGIR*, ICTIR, (New York, NY, USA), ACM, 2022.

[3] T. Ketola and T. Roelleke, "Automatic and Analytical Field Weighting for Structured Document Retrieval," in *Advances in Information Retrieval*, ECIR '23, 2023.

[4] T. Ketola and T. Roelleke, "Document structure-driven investigative information retrieval," *Information Systems*, vol. 121, p. 102315, 2024.

[5] A. Lehren, "The Rise of Investigative Data Journalism," in *Digital Investigative Journalism: Data, Visual Analytics and Innovative Methodologies in Internation Reporting*, Plagrave Macmillan, 1 ed., 2018.

[6] N. Kayser-Bril, "Measuring the Unmeasured with Data," in *Digital Investigative Journalism: Data, Visual Analytics and Innovative Methodologies in Internation Reporting*, Palgrave Macmillan, 1 ed., 2018.

[7] D. D. Le and H. W. Lauw, "Multiperspective Graph-Theoretic Similarity Measure," in *CIKM '18*, (New York, NY, USA), ACM, Oct. 2018.

[8] F. Obermaier and B. Obermayer, *Panama Papers: Breaking the Story of How the Rich and Powerful Hide Their Money.* Oneworld Publications, 2016.

[9] L. Sisti, P. Biondani, and E. Diaz-Struck, "Counting the Panama Papers money: how we reached $1.24 billion," *ICIJ*, 2019.

[10] G. Marchionini, "Exploratory search: from finding to understanding," *Communications of the ACM*, vol. 49, pp. 41–46, Apr. 2006.

[11] Y. Li and N. J. Belkin, "A faceted approach to conceptualizing tasks in information seeking," *Information Processing & Management*, vol. 44, pp. 1822–1837, Nov. 2008.

[12] R. W. White and R. A. Roth, "Exploratory Search: Beyond the Query-Response Paradigm," *Synthesis Lectures on Information Concepts, Retrieval, and Services*, vol. 1, no. 1, 2009. Morgan & Claypool Publishers.

[13] K. Athukorala, D. Głowacka, G. Jacucci, A. Oulasvirta, and J. Vreeken, "Is exploratory search different? A comparison of information search behavior for exploratory and lookup tasks," *Journal of the Association for Information Science and Technology*, vol. 67, no. 11, 2016.

[14] J. Devlin, M.-W. Chang, K. Lee, and K. Toutanova, "Bert: Pre-training of deep bidirectional transformers for language understanding," 2019.

[15] D. H. Kraft, G. Bordogna, and G. Pasi, "Fuzzy Set Techniques in Information Retrieval," in *Fuzzy Sets in Approximate Reasoning and Information Systems*, The Handbooks of Fuzzy Sets Series, Boston, MA: Springer US, 1999.

[16] G. Pasi, "Fuzzy Sets in Information Retrieval: State of the Art and Research Trends," in *Fuzzy Sets and Their Extensions: Representation, Aggregation and Models*, Studies in Fuzziness and Soft Computing, Berlin, Heidelberg: Springer, 2008.

[17] L. Page, S. Brin, R. Motwani, and T. Winograd, "The PageRank Citation Ranking: Bringing Order to the Web.," Technical Report 1999-66, Stanford InfoLab, Nov. 1999. Backup Publisher: Stanford InfoLab.

[18] S. K. M. Wong, W. Ziarko, and P. C. N. Wong, "Generalized vector spaces model in information retrieval," in *SIGIR '85*, (New York, NY, USA), ACM, June 1985.

[19] T. Roelleke, *Information Retrieval Models: Foundations and Relationships.* Morgan & Claypool Publishers, 2013.

[20] K. Spark-Jones, "A Statistical Interpretation of Term Specificity and Its Application in Retrieval," *Journal of Documentation*, vol. 28, no. 5, 1972.

[21] S. Robertson, "Understanding inverse document frequency: on theoretical arguments for IDF," *Journal of Documentation*, vol. 60, no. 5, pp. 503–520, 2004. Emerald Group Publishing Limited.

[22] S. E. Robertson and K. S. Jones, "Relevance weighting of search terms," *Journal of the American Society for Information Science*, vol. 27, no. 3, 1976.

[23] S. E. Robertson and S. Walker, "On relevance weights with little relevance information," in *SIGIR '97*, (New York, NY, USA), ACM, 1997.

[24] T. Roelleke, "A frequency-based and a poisson-based definition of the probability of being informative," in *SIGIR '03*, (New York, NY, USA), ACM, July 2003.

[25] T. Joachims, "A Probabilistic Analysis of the Rocchio Algorithm with TFIDF for Text Categorization.," in *ICML*, 1997. Section: Technical Reports.

[26] H. Fang, T. Tao, and C. Zhai, "A formal study of information retrieval heuristics," in *SIGIR '04*, (New York, NY, USA), ACM, 2004.

[27] H. Fang and C. Zhai, "An exploration of axiomatic approaches to information retrieval," in *SIGIR '05*, 2005.

[28] H. Fang, T. Tao, and C. Zhai, "Diagnostic Evaluation of Information Retrieval Models," *ACM Transactions on Information Systems*, vol. 29, no. 2, 2011.

[29] R. Cummins, J. H. Paik, and Y. Lv, "A Polya Urn Document Language Model for Improved Information Retrieval," *ACM Transactions on Information Systems*, vol. 33, no. 4, 2015.

[30] S. Balaneshinkordan, A. Kotov, and F. Nikolaev, "Attentive Neural Architecture for Ad-hoc Structured Document Retrieval," in *CIKM '18*, (Torino, Italy), ACM, Oct. 2018.

[31] O. Khattab and M. Zaharia, "ColBERT: Efficient and Effective Passage Search via Contextualized Late Interaction over BERT," June 2020. arXiv:2004.12832 [cs].

[32] A. Câmara and C. Hauff, "Diagnosing BERT with Retrieval Heuristics," in *Advances in Information Retrieval*, Lecture Notes in Computer Science, (Cham), Springer International Publishing, 2020.

[33] S. Robertson, S. Walker, S. Jones, M. M. Hancock-Beaulieu, and M. Gatford, "Okapi at TREC-3," in *Overview of the Third Text REtrieval Conference (TREC-3)*, pp. 109–126, 1995.

[34] S. E. Robertson and S. Walker, “Some Simple Effective Approximations to the 2-Poisson Model for Probabilistic Weighted Retrieval,” in *SIGIR ’94*, (London), Springer, 1994.

[35] S. E. Robertson, C. J. van Rijsbergen, and M. F. Porter, “Probabilistic models of indexing and searching,” in *SIGIR ’80*, SIGIR ’80, (GBR), pp. 35–56, Butterworth & Co., 1980.

[36] J. H. Paik, “A novel TF-IDF weighting scheme for effective ranking,” in *SIGIR ’13*, (New York, NY, USA), ACM, 2013.

[37] J. M. Ponte and W. B. Croft, “A language modeing approach to information retrieval,” in *SIGIR ’98*, (New York, NY, USA), pp. 275–281, ACM, 1998.

[38] C. Zhai and J. Lafferty, “A study of smoothing methods for language models applied to Ad Hoc information retrieval,” in *SIGIR ’01*, (New Orleans, Louisiana, USA), pp. 334–342, ACM, 2001.

[39] G. Amati and C. J. Van Rijsbergen, “Probabilistic models of information retrieval based on measuring the divergence from randomness,” *ACM Transactions on Information Systems*, vol. 20, no. 4, pp. 357–389, 2002.

[40] J. Hintikka, “On Semantic Information,” in *Physics, Logic, and History: Based on the First International Colloquium held at the University of Denver, May 16–20, 1966*, pp. 147–172, Boston, MA: Springer US, 1970.

[41] T.-Y. Liu, “Learning to Rank for Information Retrieval,” *Foundations and Trends® in Information Retrieval*, vol. 3, no. 3, pp. 225–331, 2009. Now Publishers, Inc.

[42] B. Liu, X. Lu, O. Kurland, and J. S. Culpepper, “Improving Search Effectiveness with Field-based Relevance Modeling,” in *Proceedings of the 23rd Australasian Document Computing Symposium*, ADCS ’18, (New York, NY, USA), pp. 1–4, ACM, Dec. 2018.

[43] A. Shashua and A. Levin, “Ranking with Large Margin Principle: Two Approaches,” *Advances in Neural Information Processing Systems 15*, June 2003.

[44] P. Li, C. J. C. Burges, and Q. Wu, “McRank: learning to rank using multiple classification and gradient boosting,” in *Proceedings of the 20th International Conference on Neural Information Processing Systems*, NIPS’07, (Red Hook, NY, USA), pp. 897–904, Curran Associates Inc., Dec. 2007.

[45] K. Crammer and Y. Singer, "Pranking with Ranking," in *Advances in Neural Information Processing Systems*, vol. 14, MIT Press, 2001.

[46] D. Cossock and T. Zhang, "Subset Ranking Using Regression," in *Learning Theory*, Lecture Notes in Computer Science, (Berlin, Heidelberg), pp. 605–619, Springer, 2006.

[47] R. Herbrich, T. Graepel, and K. Obermayer, "Large margin rank boundaries for ordinal regression," *Advances in Large Margin Classifiers*, vol. 88, Jan. 2000.

[48] Y. Freund, R. Iyer, R. E. Schapire, and Y. Singer, "An efficient boosting algorithm for combining preferences," *The Journal of Machine Learning Research*, vol. 4, pp. 933–969, Dec. 2003.

[49] C. Burges, "From ranknet to lambdarank to lambdamart: An overview," *Learning*, vol. 11, Jan. 2010.

[50] Y. Cao, J. Xu, T.-Y. Liu, H. Li, Y. Huang, and H.-W. Hon, "Adapting ranking SVM to document retrieval," in *SIGIR '06*, (New York, NY, USA), pp. 186–193, ACM, Aug. 2006.

[51] Z. Zheng, H. Zha, T. Zhang, O. Chapelle, K. Chen, and G. Sun, "A General Boosting Method and its Application to Learning Ranking Functions for Web Search," in *Advances in Neural Information Processing Systems*, vol. 20, Curran Associates, Inc., 2007.

[52] M.-F. Tsai, T.-Y. Liu, T. Qin, H.-H. Chen, and W.-Y. Ma, "FRank: a ranking method with fidelity loss," in *SIGIR*, pp. 383–390, 2007.

[53] Q. Wu, C. J. C. Burges, K. M. Svore, and J. Gao, "Adapting boosting for information retrieval measures," *Information Retrieval*, vol. 13, no. 3, pp. 254–270, 2010.

[54] C. Burges, R. Ragno, and Q. Le, "Learning to Rank with Nonsmooth Cost Functions," in *Advances in Neural Information Processing Systems*, vol. 19, MIT Press, 2006.

[55] Y. Yue, T. Finley, F. Radlinski, and T. Joachims, "A support vector method for optimizing average precision," in *SIGIR '07*, (New York, NY, USA), pp. 271–278, ACM, 2007.

[56] J. Xu and H. Li, "AdaRank: a boosting algorithm for information retrieval," in *SIGIR '07*, (New York, NY, USA), pp. 391–398, Association for Computing Machinery, 2007.

[57] Z. Cao, T. Qin, T.-Y. Liu, M.-F. Tsai, and H. Li, "Learning to Rank: From Pairwise Approach to Listwise Approach," in *Proceedings of the 24th International Conference on Machine Learning*, vol. 227, pp. 129–136, Jan. 2007.

[58] M. Taylor, J. Guiver, S. Robertson, and T. Minka, "SoftRank: optimizing non-smooth rank metrics," in *WSDM*, pp. 77–86, Jan. 2008.

[59] J. Xu, T.-Y. Liu, M. Lu, H. Li, and W.-Y. Ma, "Directly optimizing evaluation measures in learning to rank," in *SIGIR '08*, (New York, NY, USA), pp. 107–114, ACM, 2008.

[60] F. Xia, T.-Y. Liu, J. Wang, W. Zhang, and H. Li, "Listwise approach to learning to rank: theory and algorithm," in *ICML '08*, (New York, NY, USA), pp. 1192–1199, ACM, 2008.

[61] T. Qin, T.-Y. Liu, and H. Li, "A general approximation framework for direct optimization of information retrieval measures," *Information Retrieval*, vol. 13, pp. 375–397, Aug. 2010.

[62] M. Trabelsi, Z. Chen, B. D. Davison, and J. Heflin, "Neural ranking models for document retrieval," *Information Retrieval Journal*, vol. 24, pp. 400–444, Dec. 2021.

[63] Y. LeCun, Y. Bengio, and G. Hinton, "Deep learning," *nature*, vol. 521, no. 7553, pp. 436–444, 2015.

[64] S. Hochreiter and J. Schmidhuber, "Long Short-Term Memory," *Neural Computation*, vol. 9, pp. 1735–1780, 11 1997.

[65] K. Cho, B. van Merrienboer, Ç. Gülçehre, F. Bougares, H. Schwenk, and Y. Bengio, "Learning phrase representations using RNN encoder-decoder for statistical machine translation," *CoRR*, vol. abs/1406.1078, 2014.

[66] R. Nogueira and K. Cho, "Passage Re-ranking with BERT," Apr. 2020. arXiv:1901.04085 [cs].

[67] S. Zhuang and G. Zuccon, "TILDE: Term Independent Likelihood moDEl for Passage Re-ranking," in *SIGIR '21*, (New York, NY, USA), pp. 1483–1492, ACM, July 2021.

[68] S. Zhuang and G. Zuccon, "Fast Passage Re-ranking with Contextualized Exact Term Matching and Efficient Passage Expansion," *ArXiv*, 2021.

[69] S. Robertson, H. Zaragoza, and M. Taylor, "Simple BM25 extension to multiple weighted fields," in *CIKM '04*, (Washington, D.C., USA), pp. 42–49, ACM, 2004.

[70] H. Zaragoza, N. Craswell, M. Taylor, S. Saria, and S. Robertson, "Microsoft Cambridge at TREC–13: Web and HARD tracks," *Text Retrieval Conference*, p. 7, 2004.

[71] P. Ogilvie and J. Callan, "Combining document representations for known-item search," in *SIGIR '03*, (New York, NY, USA), ACM, 2003.

[72] A. Bookstein and R. Swanson, "Probabilistic Model for Automatic Indexing," *Journal of the American Society for Information Science*, 1974.

[73] S. Harper, "A Probabilistic Approach to Keyword Indexing," *Journal of the American Society for Information Science*, 1975.

[74] R. Wilkinson, "Effective retrieval of structured documents," in *SIGIR '94*, (Berlin, Heidelberg), pp. 311–317, Springer-Verlag, 1994.

[75] M. Lalmas, "Dempster-Shafer's Theory of Evidence applied to Structured Documents: modelling Uncertainty," *ACM SIGIR Forum*, vol. 31, Dec. 2000.

[76] M. Lalmas, "Uniform Representation of Content and Structure for Structured Document Retrieval," *Research and Development in Intelligent Systems*, July 2000.

[77] M. Lalmas and I. Ruthven, "Representing and retrieving structured documents using the Dempster-Shafer theory of evidence: modelling and evaluation," *Journal of Documentation*, vol. 54, pp. 529–565, Jan. 1998. MCB UP Ltd.

[78] G. Kazai, M. Lalmas, and T. Rölleke, "A Model for the Representation and Focussed Retrieval of Structured Documents Based on Fuzzy Aggregation.," in *Spire*, pp. 123–135, 2001.

[79] S. H. Myaeng, D.-H. Jang, M.-S. Kim, and Z.-C. Zhoo, "A flexible model for retrieval of SGML documents," in *SIGIR '98*, (New York, NY, USA), pp. 138–145, ACM, Aug. 1998.

[80] B. Piwowarski and P. Gallinari, "A Machine Learning Model for Information Retrieval with Structured Documents," in *Machine Learning and Data Mining in Pattern Recognition*, Lecture Notes in Computer Science, (Berlin, Heidelberg), pp. 425–438, Springer, 2003.

[81] C. Baumgarten, "A probabilistic model for distributed information retrieval," *ACM SIGIR Forum*, vol. 31, pp. 258–266, July 1997.

[82] M. Lalmas, T. Rölleke, and N. Fuhr, "Intelligent Retrieval of Hypermedia Documents," in *Intelligent Exploration of the Web*, Studies in Fuzziness and Soft Computing, pp. 324–344, Heidelberg: Physica-Verlag HD, 2003.

[83] M. Lalmas and T. Rölleke, "Modelling vague content and structure querying in XML retrieval with a probabilistic object-relational framework," in *International Conference on Flexible Query Answering Systems*, pp. 432–445, Springer, 2004.

[84] T. Roelleke, M. Lalmas, G. Kazai, I. Ruthven, and S. Quicker, "The Accessibility Dimension for Structured Document Retrieval," in *Advances in Information Retrieval*, Lecture Notes in Computer Science, ECIR '02, (Berlin, Heidelberg), 2002.

[85] S. Amer-Yahia and M. Lalmas, "XML search: languages, INEX and scoring," *ACM SIGMOD Record*, vol. 35, no. 4, pp. 16–23, 2006.

[86] M. Theobald, R. Schenkel, and G. Weikum, "TopX and XXL at INEX 2005," in *Advances in XML Information Retrieval and Evaluation*, Lecture Notes in Computer Science, (Berlin, Heidelberg), pp. 282–295, Springer, 2006.

[87] N. Fuhr, "An Extension of XQL for Information Retrieval," *Computer Science*, 2000.

[88] J. Kamps, M. Marx, M. de Rijke, and B. Sigurbjörnsson, "Understanding content-and-structure," in *INEX*, pp. 14–21, Citeseer, 2005.

[89] J. Kamps, M. Marx, M. d. Rijke, and B. Sigurbjörnsson, "Articulating information needs in XML query languages," *ACM Transactions on Information Systems (TOIS)*, vol. 24, no. 4, pp. 407–436, 2006. Publisher: ACM New York, NY, USA.

[90] B. Sigurbjörnsson, J. Kamps, and M. d. Rijke, "Mixture models, overlap, and structural hints in XML element retrieval," in *International Workshop of the Initiative for the Evaluation of XML Retrieval*, pp. 196–210, Springer, 2004.

[91] P. Ogilvie and J. Callan, "Language Models and Structured Document Retrieval," *INEX*, Jan. 2003.

[92] S. Robertson and H. Zaragoza, "The Probabilistic Relevance Framework: BM25 and Beyond," *Foundations and Trends® in Information Retrieval*, vol. 3, pp. 333–389, Dec. 2009.

[93] H. Zamani, B. Mitra, X. Song, N. Craswell, and S. Tiwary, “Neural Ranking Models with Multiple Document Fields,” in *WSDM ’18*, (New York, NY, USA), ACM, 2018.

[94] F. Hasibi, F. Nikolaev, C. Xiong, K. Balog, S. Bratsberg, A. Kotov, and J. Callan, “DBpedia-Entity v2: A Test Collection for Entity Search,” in *SIGIR ’17*, 2017.

[95] J. Kim, X. Xue, and W. B. Croft, “A Probabilistic Retrieval Model for Semistructured Data,” in *Advances in Information Retrieval*, Lecture Notes in Computer Science, ECIR ’2009, (Berlin, Heidelberg), pp. 228–239, Springer, 2009.

[96] K. Balog and R. Neumayer, “A test collection for entity search in DBpedia,” in *SIGIR ’13*, (New York, NY, USA), pp. 737–740, ACM, July 2013.

[97] D. Metzler and W. B. Croft, “A Markov random field model for term dependencies,” in *SIGIR ’05*, (New York, NY, USA), pp. 472–479, ACM, Aug. 2005.

[98] M. Bendersky, D. Metzler, and W. B. Croft, “Learning concept importance using a weighted dependence model,” in *WSDM ’10*, (New York, NY, USA), pp. 31–40, ACM, Feb. 2010.

[99] N. Zhiltsov, A. Kotov, and F. Nikolaev, “Fielded Sequential Dependence Model for Ad-Hoc Entity Retrieval in the Web of Data,” in *SIGIR ’15*, 2015.

[100] J. I. Choi, S. Kallumadi, B. Mitra, E. Agichtein, and F. Javed, “Semantic Product Search for Matching Structured Product Catalogs in E-Commerce,” *arXiv:2008.08180 [cs]*, Aug. 2020. arXiv: 2008.08180.

[101] H. Fang and C. Zhai, “Semantic term matching in axiomatic approaches to information retrieval,” in *SIGIR ’06*, (New York, NY, USA), ACM, 2006.

[102] D. Rennings, F. Moraes, and C. Hauff, “An Axiomatic Approach to Diagnosing Neural IR Models,” in *Advances in Information Retrieval*, ECIR ’19, 2019.

[103] A. Aizawa, “An information-theoretic perspective of tf–idf measures,” *Information Processing & Management*, vol. 39, pp. 45–65, Jan. 2003.

[104] J. Wang and T. Rölleke, "Context-Specific Frequencies and Discriminativeness for the Retrieval of Structured Documents," in *Lecture Notes in Computer Science, ECIR '06*, vol. 3936, Springer, 2006.

[105] J. Y. Kim and W. B. Croft, "A field relevance model for structured document retrieval," in *Proceedings of the 34th European conference on Advances in Information Retrieval*, ECIR'12, (Berlin, Heidelberg), pp. 97–108, Springer-Verlag, Apr. 2012.

[106] D. Metzler and W. Croft, "Linear feature-based models for information retrieval," *Inf. Retr.*, vol. 16, pp. 1–23, Jan. 2007.

[107] C. Kamphuis, A. P. de Vries, L. Boytsov, and J. Lin, "Which BM25 Do You Mean? A Large-Scale Reproducibility Study of Scoring Variants," in *Advances in Information Retrieval*, ECIR '20, pp. 28–34, 2020.

[108] N. Fuhr, "Some Common Mistakes In IR Evaluation, And How They Can Be Avoided," *ACM SIGIR Forum*, vol. 51, pp. 32–41, Feb. 2018.

[109] T. Sakai, "On Fuhr's guideline for IR evaluation," *ACM SIGIR Forum*, vol. 54, pp. 12:1–12:8, Feb. 2021.

[110] A. Trotman, A. Puurula, and B. Burgess, "Improvements to BM25 and Language Models Examined," in *Proceedings of the 2014 Australasian Document Computing Symposium*, ADCS '14, (New York, NY, USA), ACM, 2014.

[111] G. Osborne, B. Turnbull, and J. Slay, "Development of InfoVis Software for Digital Forensics," in *2012 IEEE 36th Annual Computer Software and Applications Conference Workshops*, pp. 213–217, July 2012.

[112] E. Higgins, *We Are Bellingcat: An Intelligence Agency for the People.* Bloomsbury Publishing, May 2021.

[113] D. Bamman, B. O'Connor, and N. A. Smith, "Proceedings of the 51st Annual Meeting of the Association for Computational Linguistics, pages 352–361, Sofia, Bulgaria, August 4-9 2013. c©2013 Association for Computational LinguisticsLearning Latent Personas of Film Characters," in *51st Annual Meeting of the Association for Computational Linguistics*, 2013.

[114] M. Lalmas and A. Tombros, "Evaluating XML retrieval effectiveness at INEX," *ACM SIGIR Forum*, vol. 41, pp. 40–57, June 2007.

[115] P. INGWERSEN, "Cognitive Perspectives of Information Retrieval Interaction: Eelements of a Cognitive IR Theory," *Journal of Documentation*, vol. 52, pp. 3–50, Jan. 1996. Publisher: MCB UP Ltd.

[116] I. Frommholz, B. Larsen, B. Piwowarski, M. Lalmas, P. Ingwersen, and K. van Rijsbergen, "Supporting polyrepresentation in a quantum-inspired geometrical retrieval framework," in *IIiX '10: Proceedings of the third symposium on Information interaction in context*, IIiX '10, (New Brunswick, New Jersey, USA), pp. 115–124, ACM, Aug. 2010.

[117] W. S. Cooper, "A definition of relevance for information retrieval," *Information Storage and Retrieval*, vol. 7, pp. 19–37, June 1971.

[118] K. Park, "The Nature of Relevance in Information Retrieval: An Empirical Study," *The Library Quarterly*, vol. 63, pp. 318–351, July 1993. Publisher: The University of Chicago Press.

[119] S. Mizzaro, "How many relevances in information retrieval?," *Interacting with Computers*, vol. 10, pp. 303–320, June 1998. Conference Name: Interacting with Computers.

[120] T. Berners-Lee, J. Hendler, and O. Lassila, "The Semantic Web," *Scientific American*, vol. 284, no. 5, pp. 34–43, 2001. Publisher: Scientific American, a division of Nature America, Inc.

[121] P. Ristoski and H. Paulheim, "Semantic Web in data mining and knowledge discovery: A comprehensive survey," *Journal of Web Semantics*, vol. 36, pp. 1–22, Jan. 2016.

[122] M. Teichmann, M. Weber, M. Zöllner, R. Cipolla, and R. Urtasun, "MultiNet: Real-time Joint Semantic Reasoning for Autonomous Driving," in *2018 IEEE Intelligent Vehicles Symposium (IV)*, pp. 1013–1020, June 2018.

[123] K. Li, Y. Zhang, K. Li, Y. Li, and Y. Fu, "Visual Semantic Reasoning for Image-Text Matching," in *Proceedings of the IEEE/CVF International Conference on Computer Vision*, pp. 4654–4662, 2019.

[124] B. Klimt and Y. Yang, "The Enron Corpus: A New Dataset for Email Classification Research," in *Machine Learning: ECML 2004*, Lecture Notes in Computer Science, (Berlin, Heidelberg), pp. 217–226, Springer, 2004.

[125] S. Alkhereyf and O. Rambow, "Work Hard, Play Hard: Email Classification on the Avocado and Enron Corpora," in *Proceedings of TextGraphs-11: the Workshop on Graph-based Methods for Natural Language Processing*, (Vancouver, Canada), pp. 57–65, ACM, Aug. 2017.

[126] V. VanBuren, D. Villarreal, T. A. McMillen, and A. L. Minnicks, "Enron Dataset Research: E-mail Relevance Classification," *Texas State University Faculty Publications-Computer Science*, Sept. 2009.

[127] J. Shetty and J. Adibi, "Discovering important nodes through graph entropy the case of Enron email database," in *LinkKDD '05*, (New York, NY, USA), pp. 74–81, ACM, Aug. 2005.

[128] R. Bekkerman, "Automatic Categorization of Email into Folders: Benchmark Experiments on Enron and SRI Corpora," *Computer Science Department Faculty Publication Series*, Jan. 2004.

[129] A. McCallum, X. Wang, and A. Corrada-Emmanuel, "Topic and Role Discovery in Social Networks with Experiments on Enron and Academic Email," *Journal of Artificial Intelligence Research*, vol. 30, pp. 249–272, Oct. 2007.

Index