Semantic-aware Retrieval Standards based on Dirichlet Compound Model to Rank Notifications by Level of Urgency

Mohammad Bahrani
School of Electronic Engineering and Computer Science
Queen Mary, University of London

Submitted in partial fulfilment of the requirements of the Degree of Doctor of Philosophy

June, 2023.

Declaration

I, Mohammad Bahrani, 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: Mohammad Bahrani Date: 09-06-2023

Publications and submission are listed in section 1.5.

Abstract

There is a growing number of notifications generated from a wide range of sources. However, to our knowledge, there is no well-known generalizable standard for detecting the most urgent notifications. Establishing reusable standards is crucial for applications in which the recommendation (notification) is critical due to the level of urgency and sensitivity (e.g. medical domain). To tackle this problem, this thesis aims to establish Information Retrieval (IR) standards for notification (recommendation) task by taking semantic dimensions (terms, opinions, concepts and user interaction) into consideration. The technical research contributions of this thesis include but not limited to the development of a semantic IR framework based on Dirichlet Compound Model (DCM); namely FDCM, extending FDCM to the recommendation scenario (RFDCM) and proposing novel opinion-aware ranking models. Transparency, explainability and generalizability are some benefits that the use of a mathematically well-defined solution such as DCM offers. The FDCM framework is based on a robust aggregation parameter which effectively combines the semantic retrieval scores using Query Performance Predictors (QPPs). Our experimental results confirm the effectiveness of such approach in recommendation systems and semantic retrieval. One of the main findings of this thesis is that the concept-based extension (term-only + concept-only) of FDCM consistently outperformed both terms-only and concept-only baselines concerning biomedical data. Moreover, we show that semantic IR is beneficial for collaborative filtering and therefore it could help data scientists to develop hybrid and consolidated IR systems comprising content-based and collaborative filtering aspects of recommendation.

Contents

List of Figures

List of Tables

List of Definitions

Acknowledgement

First, I would like to express my deepest gratitude to my supervisor, Thomas Roelleke, for his unfailing support, patience and kindness during all these years. He taught me many valuable lessons. As a result, we published several papers together and I have become very passionate about research and academia. I would also like to thank my family for helping me to overcome the obstacles on my way.

Chapter 1

Introduction

1.1 From traditional IR to transparent and urgency-oriented notification filtering

Traditional term-based ranking systems treat information needs as bag of words. They determine the importance of query terms regardless of semantics and are purely based on probabilities of occurrences in the collection. Although ignoring semantics such as concepts and opinions is not desirable, it would not normally lead to harmful consequences in our day-to-day tasks. The problem would arise when the accuracy of retrieval becomes significantly crucial due to the critical nature of some applications (e.g. medical and crime domains). Concerning the medical domain, notifying doctors or patients of critical health-related messages (notifications) requires a reliable filtering tool. In this context, a notification is defined as the analysis of health-related data captured from sources such as body sensors, patient profiles, narratives, lab results and hospital historical records, which is delivered in different ways (e.g push notifications). The notification filtering in the medical domain is important not only because of the need for accuracy, but also because of the special part semantics and opinions play there. For example, the smallest mistake in the determination of most urgent medical notifications would take a heavy toll on patients and healthcare organizations and therefore, developing semantic-based retrieval standards (e.g. concept-based and opinion-based) for urgent notification filtering is inevitably beneficial.
Concerning the medical domain, one of the challenges is overloading health providers and agencies with too much data which may distract them from

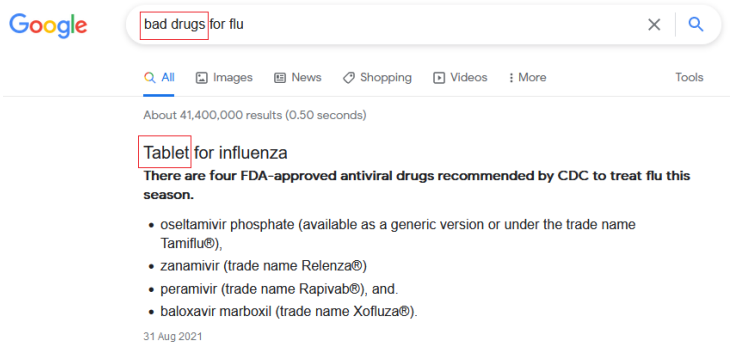
treating patients properly. Hospitals are already struggling with unutilized data, difficulty in determining prioritized health-related notifications and delivering messages to designated physicians and emergency departments. cost-effectiveness, empowered patients and better care are three significant benefits that future of health services can gain by a widespread adoption of effective urgent notification filtering. One barrier against reaching these goals is the lack of a globally accepted standard for filtering notifications generated from loads of sources. This deficiency negatively impacts health providers so that they might not trust current systems. If notification filtering is not handled effectively, urgent messages might be lost and make both patients and healthcare organizations confused. On the other hand, effective use of big data holds the promise of supporting a wide range of medical and healthcare functions, including clinical decision supports and population health management [14].
Many professionals do not trust black-box AI (Machine Learning, Neural Networks) approaches especially for critical tasks. Moreover, these approaches have their own complexities (e.g. large memory requirement). On the other hand, advanced probabilistic models are transparent and trustworthy but difficult to understand or even forgotten due to a lack of deep knowledge in the field. I propose a generalizable and transparent framework based on Information Retrieval (IR) and in particular Dirichlet Compound Model (DCM) [23]. DCM is an advanced method derived from Dirichlet-multinomial distribution [116] and therefore has a strong mathematical justification. This framework is semantically reinforced by concepts and opinions. Below we rationalize the benefits of using opinions and concepts for solving the urgency problem.

1.1.1 The need for opinion-aware IR

Sentiment analysis has been widely used in ML for many years. When it comes to urgent notification filtering through IR, it is naive to define lexical features (a group of opinions) as plain bag of words (BOW) because they have sentiment scores with different degrees of intensity. For example, let's say a notification comes with a patient narrative where the patient reports 'fatigue', 'headache', and 'feeling very bad' as their symptoms. A traditional IR approach is not able to identify the importance of the terms 'bad' and 'very' as well as their relation to the medical/psychological context. Moreover, it ignores these opinion words due to their high frequency in the collection

which in turn negatively impacts the quality of urgent notification filtering. Therefore, the consolidation of sentiment analysis with IR is beneficial for notification filtering in critical applications. Please note that there are applications where the use of opinion-based IR is theoretically more justified, especially when the decision is made based on opinions and not expert inputs (e.g. room bookings, eCommerce/online retailers, etc). The subjectivity of opinions in the medical domain is different from likes and dislikes and although the use of opinions might not be correlated to the relevance, it could play a significant role in understanding how a patient feels. Concerning the medical domain, the rational is that the more intense a lexical feature is in a query, the more influential it is to the urgency level. Moreover, the sentiment score is an exemplary factor in the determination of mental health disease for instance, there is a considerable gap between the urgency level of the query *I am feeling bad* and the query *I am feeling extremely bad* due to the high intensity of the lexical feature *extremely bad*. The BOW approach commonly treats opinions as stop-words because of their high frequency in collection. It can thus be concluded that using an opinion-based approach based on Bag of opinions is sensible to tackle this problem. Figure 1 explains better the need for such an approach. As can be seen, the provided result was not correct in relation to the given query. The snippet listed effective medications for flu which implies that the opinion word *bad* was not taken into consideration. This is because this term is treated as a stop-word due to its high frequency in collection.

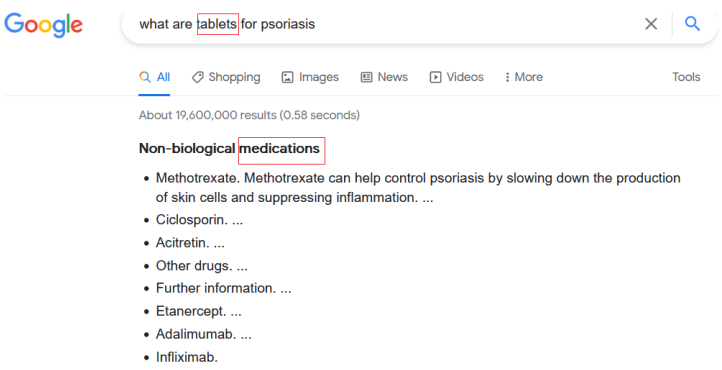
Figure 1: Results of query 'bad drugs for flu': opinion term 'bad' was ignored and wrong results for the query were shown.



1.1.2 The need for concept-based IR

A standard interpretation of query concepts (entities) including symptoms, signs and conditions is required for measuring the urgency. This is not achievable through the BOW approach and therefore we need to establish concept-based methods. This is because firstly a concept (e.g. headache) might be very frequent in collection but carries a substantial weight against the patient's overall wellbeing. Secondly, BOW is not semantic enough to build the association between a concept and other members of the concept hierarchy. In other words, BOW only processes the concept as an individual term and ignores the role of related parent and children entities in retrieval. Figure 2 demonstrates this problem by providing an example query. The expected result of the query must be a list of tablets as supposed to medications. Drugs could be categorized into various sub classes (e.g. liquids, tablets and drops). In other words, drug is a super entity (concept) while tablet is an instance of it. The wrong answer in this example shows that conceptual hierarchy was not considered wisely in the search engine.

Figure 2: Result of query 'what are tablets for psoriasis': Wrong concept is retrieved (medications instead of tablets).



1.2 Research aims (Hypotheses)

1.2.1 High level aims

1. **Establish IR-based standards for notification (recommendation) task to bridge the gap between text-based ranking and**

recommendation: Bring advanced IR to the notification scenario by developing a ranking schema for recommending items to users. Recommendation and notification filtering concepts are highly related and even convey the same meaning [11]. The rational is that using recommendation approaches such as collaborative filtering is the basis for filtering, additionally recommendation could simply imply notifying. The use of IR for notification filtering is based on effectively applying features of IR onto the recommendation scenario.

2. **Improve ranking in critical applications by taking level of urgency into consideration** Develop ranking standards based on semantics which are influential to the determination of urgency.

1.2.2 Technical aims

1. Develop a concept-only model based on Dirichlet Compound Model (DCM) by replacing terms with concepts in classic IR. Combine term-only and concept-only retrieval scores leading to a semantic standard which is discussed in section 4.4.2.

2. Develop a robust aggregation parameter (section 4.4.2) which combines individual ranking scores for semantic pillars of the corpus (concepts and terms).

3. Introduce novel opinion-based instances of traditional IR baselines (section 5.2.2) by replacing terms with opinions. This would be the result of developing a term frequency quantification derived from sentiment scores of query and document opinions.

4. Create a benchmark based on Amazon reviews of medical products (section 6.3.2) with a balanced occurrences of concepts and opinions. The aim of such a benchmark is to facilitate the research in opinion-aware retrieval for medical applications which in turn helps the research in urgent notification filtering.

5. Develop a IR-oriented recommender system based on DCM and semantics by mapping item to term and user-interaction (e.g. rating) to term frequency of traditional IR (section 6.2).

6. Train ML models using features derived from the proposed IR recommender system in order to rank similar items. This approach is discussed in section 6.2.2.

7. Apply DCM-based IR as a similarity measure in defining the neighbourhood for the KNN framework (section 6.2.2).

1.3 Evaluation

1. Evaluate and analyse the performance of different DCM-based IR approaches concerning medical applications (section 3.2.3).

2. Evaluate the quality of conceptual IR against well-established medical benchmarks such as OHSUMED and TREC using accuracy measures including MAP, Recall and Reciprocal Rank. This is discussed in section 4.4.3. Additionally, perform an individual query analysis to capture correlation between query features (e.g. type of information need) and effectiveness of conceptual IR (section 4.2.3).

3. Evaluate the performance of opinion-aware models on sentiment-based datasets (e.g reviews) and compare the results with those captured from the BOW approach (section 5.2.3).

4. Measure the capability of opinion-ware models with the polarity classification task and compare the models with a traditional sentiment analysis tool (section 5.2.3).

5. Investigate the suitability of Amazon dataset of reviews for semantic IR; specifically, opinion-aware models. This is achievable by applying semantically aggregated IR (opinions, concepts and terms) on the dataset and comparing them with the baselines (section 6.3.3).

6. Confirm that IR-based recommender systems provide robust results when applied on recommendation benchmarks (e.g. MovieLens) by using measures such as HitRate and MAP (section 6.2.3).

7. Evaluate the performance of IR-based KNN on recommendation datasets and analyse the effects of tuning the number of nearest neighbours. This analysis is included in section 6.2.3.

1.4 Glossary

1.4.1 General mathematical and IR concepts

- **TF** is Term Frequency.
- **CF** is Concept Frequency.
- **IDF** is Inverse Document Frequency.
- **BM25** is Best Match 25 (most popular ranking algorithm).
- **TLM** is Term-based Language Modelling.
- **CLM** is Concept-based Language Modelling.
- **DCM** is Dirichlet Compound Model.
- **TDCM** is Term-based Dirichlet Compound Model.
- **CDCM** is Concept-based Dirichlet Compound Model.

1.4.2 Concepts introduced in this thesis

- **OF** is Opinion Frequency.
- **IDCM** is Item-based Dirichlet Compound Model.
- **FDCM** is Full Dirichlet Compound Model.
- **RFDCM** is Recommender Full Dirichlet Compound Model.
- **URFDCM** is User-based Recommender Full Dirichlet Compound Model.
- **RFDCM$_{\mathbf{Basic}}$** is Basic Recommender Full Dirichlet Compound Model (KNN-based).
- **RFDCM$_{\mathbf{LR}}$** is Recommender Full Dirichlet Compound Model with Logistic Regression.
- **TF-IDF-sentiment** is sentiment-based Term Frequency Inverse Document Frequency.

- **TF-IDF-intense** is intensity-based Term Frequency Inverse Document Frequency.
- **LM-sentiment** is sentiment-based Language Modelling.
- **LM-intense** is intensity-based Language Modelling.

1.5 Publications and submissions

1.5.1 Publications

- **FDCM: Towards Balanced and Generalizable Concept-based Models for Effective Medical Ranking:** Mohammad Bahrani and Thomas Roelleke. 2020, In Proceedings of the 29th ACM International Conference on Information & Knowledge Management.

- **Opinion-Aware Retrieval Models Based on Sentiment and Intensity of Lexical Features:** Mohammad Bahrani and Thomas Roelleke. 2021, In Modern Management based on Big Data II and Machine Learning and Intelligent Systems III, pp. 24-31. IOS Press, 2021.

- **ADOR: A New Medical Dataset for Sentiment-based IR:** Mohammad Bahrani and Thomas Roelleke. 2021, In Proceedings of CIKM Workshops (KDAH) 2021.

- **Novel Query Performance Predictors and their Correlations for Medical Applications:** Mohammad Bahrani and Thomas Roelleke. 2018. In Proc. 1st International Workshop on Generalization in Information Retrieval (GLARE 2018).

1.5.2 Submissions

- **Semantic-aware Retrieval and Recommendation based on Dirichlet Compound Model:** Mohammad Bahrani and Thomas Roelleke. 2022, In Information Retrieval Journal.

- **From Multinomial over Dirichlet-multinomial back to Poisson:** Thomas Roelleke and Mohammad Bahrani. 2022, To be submitted in Information Processing & Management Journal.

Chapter 2

Related Work

2.1 Semantic retrieval

A fundamental angle of retrieval is semantic representation of data. Conceptual graphs, ontological knowledge-based systems are well-known traditional examples of semantic retrieval methods which have been popular from long time ago. In 1968, [37] studied indexing semantics including single words, collections of words, or syntactic phrases. Later, [25], discussed the latent semantic indexing (LSI) analysis to improve ranking by constructing a semantic space from the matrix of term-document association data. In 1992, [77] showed that integration of semantics and probabilistic approaches is feasible. Later, [72] proposed a semantic framework based on terminological logic. POLAR is an interesting probabilistic framework for annotation-based retrieval introduced by [32], it is object-oriented and created based on characteristics of annotations (objects), documents and their relationships. XML-based schema has also been very popular among researchers for representation, e.g. [6] proposed some semantic retrieval models driven from a generic knowledge-oriented schema for semantically expressing queries using XML. Their proposed Semantic Query Rating Scheme (SQR) determines the semantic complexity of the query by mapping query processing methods for each level of interpretation. In another work on the semantic complexity of queries, [21] proposed an ontology-based framework that semantically annotates documents within a retrieval system.
Researchers introduced concept-based IR to address two vocabulary problems of the BOW approach commonly known as polysemy and synonymy

[27]. In retrieval, synonymy would appear when the words (terms) chosen by users are different from those used by experts and polysemy is the result of ambiguous terms in documents and queries with different contextual meanings. Concept-based IR leverages semantic representations of documents and queries based on concepts (or in addition) to terms in a conceptual space. There have been many studies on concept-based retrieval. For example, [74] proposed and investigated concept-based language models for domain specific retrieval. Later, [109] reinforced the plain conceptual language model by the consideration of semantic types and their importances. [65] proposed a task-specific query and document representation by focusing only on the medical concepts and their implicit relationships. [64] and [33] leveraged pattern-based topics for filtering documents. Later, [109] reinforced the plain concept-based LM by using the semantic types and the level of importance. [95] presented a joint model for Named entity Recognition (NER) and Entity Linking (EL); namely, Re-Ranking. [1] used a rule-based approach to allow users to explore the annotated medical relations in RDF format.
Unlike traditional semantic retrieval approaches, neural networks and ML approaches implicitly cover semantic aspects since they typically model a term as a vector. Neural networks are widely used in entity and relation extraction studies, e.g. [3] developed a sentence-level Bidirectional Long Short-Term Memory (BiLSTM) neural network for biomedical retrieval by exploiting structured information extraction. The TREC 2012 Medical records Track allowed the electronic health records to be retrieved based on the semantic features of the fields [108].

2.2 Opinion-aware retrieval

A wide range of research has been done on the polarity classification of textual data using machine learning. Companies need to analyse customer's general feelings about their products. On the other hand, singular buyers need to know the sentiment of the product reviews before buying [113]. Wherefore the examination of sentiments would be beneficial for many applications. Researchers need to analyse the sentiment intensity over time to know about changes in rhetoric [114]. This would benefit analysts in companies, hospitals, government and political departments that need to track emotions and attitudes. To date, sentiment analysis is mostly applied to the polarity classification task, however this may not be sufficient for many

domains. Companies need to provide people with search engines that are able to retrieve top products based on user queries and sentiment analysis of reviews. This could help users to match their inputs with products that have the best reviews, however traditional IR is not intelligent enough for this purpose. This is because it does not capture opinions and treats them like plain terms. For example, the opinion word 'good' might occur nearly in every positive review and if a user searches for 'good tablets', the traditional system would not consider the term 'good' informative and selective and consequently, could not rank the results based on sentiment polarity.
[113] applied sentiment classification techniques including Naive Bayes and SVM on movie reviews and showed that Naive Bayes is more effective than SVM. [36] used distant supervision methods to classify twitter messages in positive or negative categories based on their sentiments. The proposed framework was to enable customers to research sentiments of products before purchase and to help companies with the acquisition of the public sentiment of their brands. They reported 82.7% accuracy for Naive Bayes considering both unigrams and bigrams. Furthermore, [79] showed that standard ML algorithms outperform human-produced sentiment labels regarding movie reviews. [96] worked on an aggregation tool built upon linguistic features of texts to assign sentiment labels to reviews.
In another study, [56] examined the improving effects of valence shifters such as negations, intensifiers, and diminishers on sentiment analysis.
To the best of our knowledge, no prior work has been done on intensity-aware IR. Interpretation of query intent is a commonly used approach to capture the information needs, but sentiments are not explicitly considered in IR. However, there is prior research on opinion retrieval and sentiment summary. Some of the relevant past papers on opinion retrieval can be found in [120, 35, 42, 51]. Bonzanini et al. [16, 17] investigated the role of summarisation extraction in the detection of sentiments. They discussed the importance of short passages where authors describe their overall feelings about movie reviews. In a similar study, [10] conducted experiments on sentence's location and constituent words as predictors in sentiment-summary discovery.
VADER (Valence Aware Dictionary for sEntiment Reasoning) is our baseline for sentiment classification task in this thesis. It is a parsimonious rule-based tool for sentiment analysis concerning social media texts developed by Hutto and Gilbert [53]. It leverages a combination of qualitative and empirical methods by the use of human experts and judgmental evaluations. Moreover, it employs a rich intensity-based lexicon to assign sentiments to

sentences. The results of their experiments were encouraging since VADER in most cases provided higher accuracy than eleven other highly regarded machine learning approaches.

2.3 IR-based Recommendation

Recommendation systems could leverage AI to deliver top-ranked notifications to third parties. There is an inherent connection between recommendation algorithms and retrieval algorithms: for the content-based side of recommendation, the retrieval algorithm contributes the similarity measure between items/products, and for the collaborative side, the similarity between users/customers. IR evaluation metrics, specifically precision, have been also reported to be robust when applied onto the top-N recommendation task (since error-based evaluation metrics are insufficient) [103, 12], and therefore, it makes the use of IR in recommendation more sensible.
The integration of IR with recommendation, specifically collaborative filtering, was well-studied in [13]. They discussed the application of well-established IR methods including TF-IDF, language modelling (LM) and BM25 in collaborative filtering and introduced an approach which is usable for any text-based weighting scheme. It consists of two steps. In the first step, products are assumed to be a list of ratings and users a lists of similar users, in the second step they swapped the way user and product lists are generated. In another work [111], demonstrated that top-N recommendation-oriented IR is effective on collections where user interaction is implicit. [105] confirmed that query likelihood is more reliable than cosine similarity in terms of retrieving similar users. In another study, [104] demonstrated that LM smoothing plays an important role in improving collaborative filtering. [100] proposed an improved language model and topic model based on terms and semantics to assign (recommend) manuscripts to reviewers. Additionally, the IR-based recommendation is discussed in [102, 110]. The research on KNN techniques in recommender systems is trendy. Relevant to our study is [92]. In this work, Cruzado et al. explored the advantages that IR could bring to contact recommendation concerning social networks. They confirmed that IR adoption is more effective than plain KNN for determining neighbourhood clusters. They also examined the integration of IR models in contact recommendation methods [91]. Regarding recent papers on improving KNN for recommender systems, we can refer to [18, 80, 61].

[18] discussed the benefits of using maximum-minimum distance algorithm for movie recommendation and [80] proposed a Hybrid Action-Related K-Nearest Neighbour (HAR-KNN) recommender system. This system is based on user behaviour matrix and product frequency. In another work, Li et al. [61] proposed an improved KNN via compression and global effect.
Collaborative memory networks (CMN) [26] combines memory networks and neural architecture mechanisms with neighbourhood approaches. Meta path-based context for Recommendation (NFR) was presented by [49] at KDD 2018. Their framework comprises a priority-based sampling approach which leverages from semantics, e.g. movie genres. [63] developed a system called Variational Autoencoders for Collaborative Filtering (Mult-VAE). Their framework is based on multinomial likelihood and can estimate the parameters using Bayesian inference.

Chapter 3

DCM-based IR

3.1 Chapter overview

This chapter focuses on investigating and establishing Dirichlet-multinomial retrieval. Particular attention goes to comparing the document-generating model against the query-generating model.
The results of this chapter include the formulation, interpretation and comparison of scores. The experimental study shows a particular Dirichlet-multinomial approach to be a superior model. The best-performing Dirichlet-multinomial approach is a Poisson model. The chapter provides formal proof regarding the rank equivalence. Moreover, the chapter contributes insights into why document-generating models perform poorer than query-generating models, if the mathematical framework is applied without adjustments regarding the term frequency.

3.2 Foundations of DCM in retrieval

3.2.1 Background

While multinomial LM (language modelling) is well-established since the late 90s, the Dirichlet-multinomial model has been proposed in 2008, and even recent research has not yet fully established the approach. In this section I introduce multinomial LM and BM25 as established IR ranking models.

Multinomial LM: D2Q: P(q|d)

LM is based on the query log-likelihood ratio of posterior $P(q|d,c)$ and prior $P(q|c)$, where the query q is the target, the collection c is the background model, and the document d is the foreground model. Please note that in this thesis, D2Q is for query-generating models and Q2D is for document-generating models. Let $n(t,q)$ be the within-query term frequency, for the log-likelihood ratio, the application of the multinomial model is:

$$\text{score}_{\text{LM}}(d,q,c) := \log \frac{P(q|d,c)}{P(q|c)} = \sum_t n(t,q)\ \log \frac{P(t|d,c)}{P(t|c)} \tag{1}$$

The term probability $P(t|d,c)$ is estimated via a mixture (weighted average) of foreground $P(t|d)$ and background probability or probability of term in the collection $P(t|c)$. The probabilities are usually based on the event counts. For example, let $n(t,d)$ be the within-document term frequency and $n(t,c)$ be the within-collection term frequency, then $P(t|d) = n(t,d)/\text{len}(d)$ and $P(t|c) = n(t,c)/\text{len}(c)$, respectively.

BM25-TF-IDF (BM25 without relevance)

LM is a query-generating model, while BM25 is a document-generating model. BM25-TF-IDF is BM25 without relevance information. Details of BM25 are not important for this thesis, but for reproducibility and for the proof regarding BM25-TF-IDF being a special case of Poisson, it is important to clarify what exactly is the TF and IDF applied.

Score Let K_{piv} denote the TF pivotisation factor, $\text{len}(d)$ be the document length, $\text{avgdl}(c)$ be the average document length in the collection, $\text{df}(t,c)$ be the document frequency of term t and N be the number of documents in the collection:

$$K_{\text{piv},b}(d,c) := b\ \text{len}(d)/\text{avgdl}(c) + (1-b) \tag{2}$$

Then, the BM25-TF is:

$$\text{TF}_{\text{BM25}}(t,d,c) := \frac{(k_1+1)\ n(t,d)}{n(t,d) + k_1\ K_{\text{piv},b}(d,c)} \tag{3}$$

The BM25-IDF is:

$$\text{IDF}_{\text{BM25}}(t,c) := \log \frac{N - \text{df}(t,c) + 0.5}{\text{df}(t,c) + 0.5} \tag{4}$$

The score is:

$$\text{score}_{\text{BM25-TF-IDF}}(d, q, c) := \sum_{t \in q} \text{TF}_{\text{BM25}}(t, d, c)\ \text{IDF}_{\text{BM25}}(t, c) \tag{5}$$

The sum is restricted to query terms, $t \in q$. The score can be also formulated with $\text{TF}(t, q, c)$, which is equivalent to restricting the sum for queries with a binary representation, $n(t, q) = 1$ or 0.

Parameter Estimation The parameters k_1 and b control the saturation of the TF weight, and the impact of the document length normalisation. For the experiments, we will consider a common setting: $k_1 = 1.2$ and $b = 0.8$ [88].

3.2.2 Contribution

Problem

A core ingredient of language modelling (LM) is mixing foreground parameters with background parameters. Let d denote a document (foreground model), and c a collection (background, language). For the multinomial LM, the mixture of foreground and background term probabilities is:

$$P(t|d, c) = w_d\ P(t|d) + (1 - w_d)\ P(t|c) \tag{6}$$

The main proposal [23] is to replace the probabilistic mixture by a mixture of Dirichlet parameters. Let $\beta_t(d)$ denote the parameter of the foreground model, and $\beta_t(c)$ be for the background. The mixture is:

$$\beta_t(d, c) = w_d\ \beta_t(d) + (1 - w_d)\ \beta_t(c) \tag{7}$$

The main contribution of [23] is to propose and investigate estimates for $\beta_t(d)$ and $\beta_t(c)$. Replacing the probabilistic mixture in the multinomial approach by a mixture of Dirichlet parameters (those parameters are based on averages) requires a mathematical justification. From a mathematical point of view, there are two options:

1. Apply the Dirichlet-multinomial distribution instead of the multinomial approach [116].

2. Remain with the multinomial approach for the term weight [23], but explain what justifies replacing probabilities by averages.

Main Contribution

The study reported in this chapter showed that the Dirichlet-multinomial approach for $P(q|d)$ can be reduced to a Poisson model. This leads to a Poisson-like formulation of what is known in IR as the multinomial LM approach. We obtain a Poisson approach, with two scores, Poisson D2Q (query-generating) and Poisson Q2D (document-generating). Poisson D2Q will be shown to be equivalent to the Dirichlet-multinomial LM approach, and Poisson Q2D has several variations where the TF component $\text{TF}(t, d, c)$ is important. Also, it can be shown to be equivalent to BM25-TF-IDF. The experiments show that those Poisson-based models outperform baselines for most measures over three standard collections.

From Multinomial to Poisson

Dirichlet-multinomial LM: D2Q: P(q|d) For Dirichlet-multinomial, the case is more complicated since the event count ($n(t, q)$ for query likelihood) is an argument of the Gamma function. [28, 116] show the reduction and application of the DCM for document classification and retrieval.
For this chapter important is to highlight the dual forms, namely the Dirichlet-multinomial for query-generating *and* document-generating models. For the query-generating (D2Q) side, the formula is justified in [116]:

Definition .1 (DCM D2Q)

$$\log \frac{P(q|d,c)}{P(q|c)} = \left(\sum_t \log \frac{\Gamma(\beta_t(d,c) + n(t,q))\ \Gamma(\beta_t(c))}{\Gamma(\beta_t(d,c))\ \Gamma(\beta_t(c) + n(t,q))} \right) - \log \frac{\Gamma(m(d,c) + n)\ \Gamma(m(c))}{\Gamma(m(d,c))\ \Gamma(m(c) + n)} \tag{8}$$

Where $n(t, q)$ is the total count of term t in query q, and β_t are the Dirichlet parameters: $\beta_t(d, c)$ for the mix of foreground and background, and $\beta_t(c)$ for the background. The sum $n = \sum_t n(t, q)$ is the query length, and $m(d, c) := \sum_t \beta_t(d, c)$ and $m(c) := \sum_t \beta_t(c)$ is the sum of Dirichlet parameters.

This chapter will highlight that the Dirichlet-multinomial D2Q approach can be shown to reduce to the Poisson D2Q score for the case of a binary target vector ($n(t,q) = 1$ for query terms, and $n(t,q) = 0$ otherwise).

Dirichlet-multinomial Q2D: $P(d|q)$ For the document-generating side [116], eq (8) becomes:

Definition .2 (DCM Q2D)

$$\log \frac{P(d|q,c)}{P(d|c)} = \left(\sum_t \log \frac{\Gamma(\beta_t(q,c) + n(t,d))\ \Gamma(\beta_t(c))}{\Gamma(\beta_t(q,c))\ \Gamma(\beta_t(c) + n(t,d))} \right) - \log \frac{\Gamma(m(q,c) + n)\ \Gamma(m(c))}{\Gamma(m(q,c))\ \Gamma(m(c) + n)} \tag{9}$$

Where $n = \sum_t n(t,d)$ is the document length, and $m(q,c) = \sum_t \beta_t(q,c)$ is the sum of the Dirichlet parameters for the foreground model, the query q. Figure 3 shows curves for illustrating the dynamics of the term weights for query-generating (D2Q) and document-generating (Q2D) Dirichlet-multinomial.
On the D2Q side, there is a smooth rise of the Dirichlet-multinomial term weight (because the term frequency $n(t,d)$ affects $\beta_t(d)$), while on the Q2D side, the term frequency does not have a strong impact (because $n(t,d)$ is an argument of the Gamma function for foreground and background model). The respective reference models, LM for D2Q and BM25-TF-IDF for Q2D, emphasise that the term weight dynamics of D2Q Dirichlet-multinomial is similar to LM, whereas Q2D Dirichlet-multinomial is rather different from the rise and saturation of the BM25-TF-IDF term weight. The experiments will underline that the retrieval quality of Q2D Dirichlet-multinomial is poor compared to BM25-TF-IDF.

Poisson LM: D2Q: P(q|d) When applying the Poisson distribution, then the query log-likelihood ratio is:

$$\log \frac{P(q|d,c)}{P(q|c)} = \sum_t \left(n(t,q)\ \log \frac{\lambda_t(d,c)}{\lambda_t(c)} + \lambda_t(c) - \lambda_t(d,c) \right) \tag{10}$$

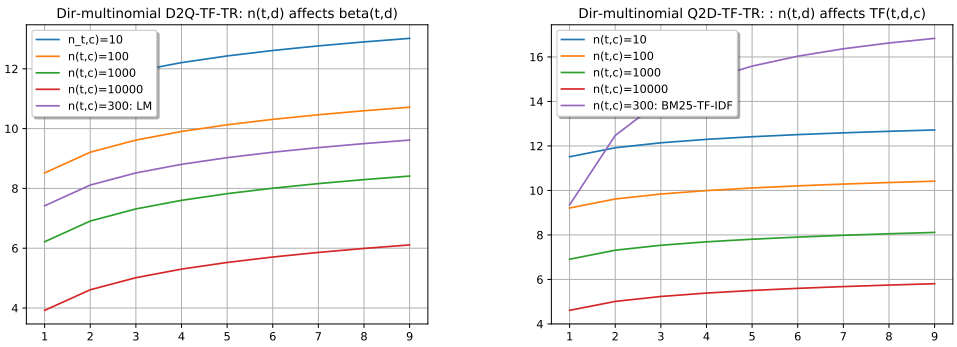


Figure 3: Dirichlet-multinomial term weights: Query-generating (left side, D2Q, $P(q|d)$) versus document-generating (right side, Q2D, $P(d|q)$): x-axis is $n(t,d)$, the number of term occurrences; y-axis is the term weight from eq (8) and eq (9), respectively; curves are for terms of varying rareness: $n(t,c)$ is the total number of occurrences of term t in collection c. Whereas on the D2Q side, the saturation of the Dirichlet-multinomial term weights fit the reference model (LM, for a term with $n(t,c) = 300$ occurrences), on the Q2D side, the saturation is much stronger than for the reference model (BM25-TF-IDF term weights).

Where $\lambda_t()$ denotes the average number of occurrences. For understanding this ratio, reconsider that $k \log(\lambda) - \lambda - \log(k!)$ is the logarithm of the Poisson probability, and the difference between posterior and prior ($\lambda_t(d,c)$ for the posterior, $\lambda_t(c)$ for the prior) leads to the log-likelihood ratio.
An important mathematical aspect is that we may drop the difference between averages if the averages are small (close to zero).

$$\log \frac{P(q|d,c)}{P(q|c)} \approx \sum_t n(t,q) \ \log \frac{\lambda_t(d,c)}{\lambda_t(c)} \tag{11}$$

It is the logarithm of the fraction between averages that dominates the score, where $\lambda_t(d,c) - \lambda_t(c) \approx 0$. Note that this can alway be achieved since a constant scaling paramter can be applied to the averages, and this does not affect the fraction of averages.
The average $\lambda_t(d,c)$ is a mixture of foreground model d and background model c:

$$\lambda_t(d,c) = w_d \ \lambda_t(d) + (1 - w_d) \ \lambda_t(c) \tag{12}$$

The Poisson model discussed here is not an established retrieval model, though the model is not new. One of the main contributions of this chapter will be to prove formally under which conditions the Poisson D2Q score is equivalent to the Dirichlet-multinomial D2Q score (proof 3.2.2).

Poisson Q2D: P(d|q) For the document log-likelihood ratio, the respective expression is:

$$\log \frac{P(d|q,c)}{P(d|c)} = \sum_t n(t,d) \ \log \frac{\lambda_t(q,c)}{\lambda_t(c)} \tag{13}$$

The model does not work well because of the total count $n(t,d)$.

Poisson Retrieval Scores Replacing the total count by a TF quantification leads to two retrieval scores that are currently not established as IR models. To reduce the impact of multiple occurrences of the same term, a saturating function [88] $\text{TF}(t,d,c) := \frac{n(t,d)}{n(t,d)+k}$ can be used for Q2D model where k is the saturating parameter. For D2Q (query-generating), replacing $n(t,q)$ by a saturating TF is less important than for Q2D, if queries are short and do not have multiple occurrences of terms ($n(t,q) = 1$ or $n(t,q) = 0$). Consequently, we can say $\text{TF}(t,d,c) := n(t,q)$.

Definition .3 (Poisson D2Q Score)

$$\text{score}_{\text{Poisson-D2Q}}(q, d, c) := \sum_t \text{TF}(t, q, c) \ \log \frac{\lambda_t(d, c)}{\lambda_t(c)} \tag{14}$$

Definition .4 (Poisson Q2D Score)

$$\text{score}_{\text{Poisson-Q2D}}(d, q, c) := \sum_t \text{TF}(t, d, c) \ \log \frac{\lambda_t(q, c)}{\lambda_t(c)} \tag{15}$$

The main contributions of this chapter are (1) to show the conditions for which Poisson-D2Q is the same as Dirichlet-multinomial, and (2) to show where Poisson-Q2D and BM25-TF-IDF meet.
For Q2D (document-generating model), replacing $n(t, d)$ by $\text{TF}(t, d, c)$ is very important, since high total counts tend to over-power the impact of other query terms.
This analysis shows why D2Q models are often shown to be superior to Q2D models, since for the latter, a term saturation and a non-query-term assumption need to be considered.

Q2D: Non-query term assumption For query-generating models (D2Q, LM), it is clear that a penalty for "a query term missing in the document" is a desirable feature/axiom for a retrieval model [31]. For document-generating models (Q2D), the corresponding approach means a penalty for "a document term missing in the query". Whereas for D2Q, the penalty works because there are few query terms against many document terms, for Q2D, the penalty is problematic because there are many document terms against few query terms. Therefore, for Q2D models (e.g. BM25), a non-query term assumption is required for removing the detrimental effect of non-query terms. Mathematically, this means that a sum $\sum_t$ over all terms will be replaced by $\sum_{t \in q}$.
For multinomial and Poisson, the respective non-query term assumption means:

$$\sum_t \text{TF}(t, d, c)\text{TR}(t, q, c) \stackrel{\text{rank}}{=} \sum_{t \in q} \text{TF}(t, d, c)\text{TR}(t, q, c) \tag{16}$$

Where the TR (term-relevance) weight is based on the term probabilities $P(t|q)$ and $P(t|c)$ for multinomial, and on the term averages $\lambda_t(q)$ and $\lambda_t(c)$ for Poisson. The non-query term assumption means $P(t|q,c) = P(t|c)$ for multinomial, and $\lambda_t(q,c) = \lambda_t(c)$ for Poisson.
Regarding the Dirichlet-multinomial case, there is no TF-TR product, since the term frequency is an argument of the Gamma function. When assuming $\beta_t(q,c) = \beta_t(c)$ for non-query terms, then the sum in eq (8) reduces to the sum over $t \in q$.

Dirichlet-multinomial: sumlog We consider next the sumlog-based formulation of fractions of Gamma expressions. First and foremost, this mathematical feature is required for comparing TF quantifications on the mathematical level. Secondly, the special case $\Gamma(\beta+1)/\Gamma(\beta) = \beta$ is important for the relationship between Dirichlet-multinomial and Poisson.
$\Gamma(b+k) = (b+k-1)\cdot\Gamma(b+k-1)$ and a fraction of Gamma expressions of the form $\Gamma(b+k)/\Gamma(b)$ can be expressed as the product $\prod_{j=0}^{k-1}(b+j)$ which leads to the sum-log-base term weight quantification in eq (19). Therefore, the Dirichlet-multinomial, eq (8), can be expressed as:

$$\log\frac{P(q|d,c)}{P(q|c)} = \left(\sum_t \sum_{j=0}^{n(t,q)-1} \log\frac{j+\beta_t(d,c)}{j+\beta_t(c)}\right) - \sum_{j=0}^{n-1}\log\frac{j+m(d,c)}{j+m(c)} \tag{17}$$

Where $n = \sum_t n(t,q)$ is the query length. It is worthwhile to consider the dual formulation for Q2D, even though only the role of target and foreground model change:

$$\log\frac{P(d|q,c)}{P(d|c)} = \left(\sum_t \sum_{j=0}^{n(t,d)-1} \log\frac{j+\beta_t(q,c)}{j+\beta_t(c)}\right) - \sum_{j=0}^{n-1}\log\frac{j+m(q,c)}{j+m(c)} \tag{18}$$

This is the sumlog formulation of eq (9) [116].

Interpretation of Term Weights The sum-log-based formulation of the Dirichlet-multinomial allows for interpreting what is the term weight in the Dirichlet-multinomial, multinomial and Poisson approach.

Without loss of generality, for a term t that occurs $n(t,q)$ times, the term weight is:

$$w_{\text{Dirichlet-multinomial}}(t,q,d,c) = \sum_{j=0}^{n(t,q)-1} \log(j + \beta_t(d,c)) \tag{19}$$

Where $\beta_t(d,c)$ is the mixture of foreground and background model.
For the multinomial D2Q approach, the term weight is:

$$w_{\text{multinomial}}(t,q,d,c) = n(t,q)\log(P(t|d,c)) \tag{20}$$

And finally, for Poisson, it is:

$$w_{\text{Poisson}}(t,q,d,c) = n(t,q)\log(\lambda_t(d,c)) \tag{21}$$

For the D2Q side of retrieval, the within-document term frequency is embedded into the estimation of the foreground parameters: $\beta_t(d,c)$, $P(t|d,c)$, or $\lambda_t(t,c)$, respectively.
For document-generating models, there is a more direct impact of the term frequency $n(t,d)$, while the query q is the foreground model.

$$w_{\text{Dirichlet-multinomial}}(t,d,q,c) = \sum_{j=0}^{n(t,d)-1} \log(j + \beta_t(q,c)) \tag{22}$$

$$w_{\text{multinomial}}(t,q,d,c) = n(t,d)\log(P(t|q,c)) \tag{23}$$

$$w_{\text{Poisson}}(t,q,d,c) = n(t,d)\log(\lambda_t(q,c)) \tag{24}$$

Scores and parameter estimation

Multinomial LM

Score Following eq (1), the LM score is:

$$\text{score}_{\text{LM}}(d,q,c) := \sum_t \text{TF}(t,q)\ \log\left(w_d\ \frac{P(t|d,c)}{P(t|c)} + (1-w_d)\right) \tag{25}$$

Parameter Estimation For estimating the collection-wide term probability, there are two approaches:

$$\text{occurrence-based: } P(t|c) = \frac{n(t,c)}{\sum_t n(t,c)} \tag{26}$$

$$\text{df-based: } P(t|c) = P_{\text{df}}(t|c) = \frac{\text{df}(t,c)}{\sum_t \text{df}(t,c)} \tag{27}$$

We will refer to the respective models as LM1 or simply LM (for occurrence-based) and LM2 or LM_{df} (for document-frequency-based).

LM: Mixture With mixture for $P(t|d,c)$, eq (25) becomes:

$$\log \frac{P(q|d,c)}{P(q|c)} = \sum_t n(t,q) \log \frac{w_d\, P(t|d) + (1-w_d)\, P(t|c)}{P(t|c)} \tag{28}$$

Where $n(t,q) = \text{TF}(t,q)$ is the within-query term frequency. This is the standard LM score [118, 47].

Cummins Dirichlet-multinomial LM

There are two break-through in [23], one regarding the multinomial score, and one regarding parameter estimation.

Score In analogy to eq (25), the so-called Dirichlet-multinomial LM score is:

$$\text{score}_{\text{Cummins-LM}}(d,q,c) := \sum_t \text{TF}(t,q) \log \left(w_d \frac{a_d(t)}{a_c(t)} + (1-w_d) \right) \tag{29}$$

Where $a_d(t)$ and $a_c(t)$ are the Dirichlet parameters, and w_d is the mixture parameter.[1] [2]

[1] We employ w_d instead of λ_d since λ denotes the parameter of the Poisson distribution.

[2] We show Cummins' notation, $a_d(t)$; on our notation, $\beta_t(d) = a_d(t)$; both notations serve the purpose, for our work, $\beta_t(d)$ is more convenient since it aligns with $\lambda_t(d)$ for the Poisson model.

Parameter Estimation Regarding the parameter estimation, let $|\overline{d}| = N_T(d)$ be the number of *distinct* terms in document d; let μ_c be the average number of distinct terms in a document of average length, i.e. $\mu_c = |\overline{d}_{\text{avg}}| = N_T(d_{\text{avg}})$, the Dirichlet parameters of foreground model d and background model c are:

$$\beta_t(d) := a_d(t) := N_T(d)\ \frac{n(t,d)}{|d|} = N_T(d)\ P(t|d) \tag{30}$$

$$\beta_t(c) := a_c(t) := \mu_c\ \frac{n(t,c)}{|c|} = \mu_c\ P(t|c) \tag{31}$$

The mixture is:

$$\beta_t(d,c) = w_d\ \beta_t(d) + (1 - w_d)\ \beta_t(c) \tag{32}$$

There are three main issues:

1. How to estimate μ_c?
2. How to estimate $P(t|c)$?
3. How to set the mixture parameter w_d?

We briefly revisit the proposals from [23].

Distinct terms in average document: μ_c For a set of n documents, the idea is that μ_c can be estimated via a recursive equation:

$$\mu_c = \frac{\sum_{i=1}^{n} |\overline{d_i}|}{\sum_{i=1}^{n} \psi\left(|d_i| + \mu_c\right) - n \cdot \psi\left(\mu_c\right)} \tag{33}$$

Where ψ is the Di-Gamma function. This equation replaces intuitive approaches such as estimating the number of distinct terms in an average document via simple counts, e.g. based on the length of the collection (number of words) and the number of documents.

Term probability $P(t|c)$ The two main options are occurrence-based and df-based. The occurrence-based estimation is:

$$P(t|c) = \frac{n(t,c)}{|c|} \tag{34}$$

Where $|c| = N_L(c)$ is the length of the collection. Alternatively, the df-based estimate is:

$$P(t|c) = P_{\text{df}}(t|c) = \frac{\text{df}(t,c)}{\sum_i \text{df}(t_i,c)} \tag{35}$$

Mixture parameter: w_d Regarding the mixture parameter, we follow [23], namely the usual approach of multinomial LM, where the mixture parameter is either a constant (JM smoothing) or proportional to the length ratio between candidate document and an average document (Dirichlet smoothing). For the experiments, we apply JM smoothing with $w_d = 0.8$.

Proofs

Dirichlet-multinomial and Poisson D2Q

This section shows a formal proof regarding the conditions that reduce the query-generating Dirichlet-multinomial approach to the Poisson approach.

Dirichlet-multinomial and Poisson. For a binary target, the log-likelihood ratio of the Dirichlet-multinomial model is equal to the Poisson model.

Proof. Dirichlet-multinomial and Poisson To show:

$$\text{score}_{\text{Dirichlet-multinomial-LM}}(d,q,c) = \log \frac{P_{\text{Poisson},\lambda}(\vec{k}|d,c)}{P_{\text{Poisson},\lambda}(\vec{k}|c)} \tag{36}$$

Where $\vec{k}$ is a vector of query term frequencies, i.e. $n(t,q)$ are the respective components of the vector.
For a binary target where $n(t,q) = 1$ or $n(t,q) = 0$, the fraction of Gamma expressions can be reduced. For $n(t,q) = 1$:

$$\frac{\Gamma(\beta+1)}{\Gamma(\beta)} = \frac{\beta\,\Gamma(\beta)}{\Gamma(\beta)} = \beta \tag{37}$$

For $n(t,q) = 0$, $\Gamma(\beta)/\Gamma(\beta) = 1$.

Then, the Dirichlet-multinomial score, eq (8), reduces to:

$$\text{score}_{\text{Dirichlet-multinomial-LM}}(d, q, c) = \left(\sum_{t \in q} \log \frac{\beta_t(d, c)}{\beta_t(c)} \right) - \log \frac{\Gamma(m(d, c) + n)\ \Gamma(m(c))}{\Gamma(m(d, c))\ \Gamma(m(c) + n)} \tag{38}$$

Note that this makes explicit that from a mathematical point of view, there is no factor $\text{TF}(t, q)$. The sum is only over the query terms, $t \in q$.
For the ranking equivalence between Dirichlet-multinomial and Poisson, the following assumption is essential.

$$\log \frac{\Gamma(m(d, c) + n)\ \Gamma(m(c))}{\Gamma(m(d, c))\ \Gamma(m(c) + n)} \propto \lambda_t(d, c) - \lambda_t(c) \tag{39}$$

For small averages, both sides converge to zero. Therefore, the score is rank-equivalent to the sum over the logarithm of the fractions of Dirichlet parameters:

$$\text{score}_{\text{Dirichlet-multinomial-LM}}(d, q, c) \overset{\text{rank}}{=} \sum_{t \in q} \log \frac{\beta_t(d, c)}{\beta_t(c)} \tag{40}$$

The Poisson score is:

$$\text{score}_{\text{Poisson}}(d, q, c) = \sum_{t} \text{TF}(t, q, c)\ \log \frac{\lambda_t(d, c)}{\lambda_t(c)} \tag{41}$$

For a binary target, where $\text{TF}(t, q, c) = 1$ or zero, the sum can be expressed over $t \in q$.

$$\text{score}_{\text{Poisson}}(d, q, c) = \sum_{t \in q} \log \frac{\lambda_t(d, c)}{\lambda_t(c)} \tag{42}$$

Then, for $\lambda_t(d, c)/\lambda_t(c) = \beta_t(d, c)/\beta_t(c)$, the Poisson score is rank-equal to the Dirichlet-multinomial score.

Note that only the ratios need to be equal, i.e. any scale parameter ϱ can be applied between average λ and Dirichlet parameter β.

$$\beta_t(d, c) = \rho\ \lambda_t(d, c) \tag{43}$$

This is important in the sense that one does not want to constraint the value range of the Dirichlet parameters and averages, apart from the fact that both of them are relatively small.
An important result of this analysis regarding the Dirichlet-multinomial model and the Poisson model is that Cummins' Dirichlet-multinomial approach is to be explained as a product of Poisson probabilities. This relationship between DCM and Poisson holds for any application of the Dirichlet-multinomial where the target is a binary vector. It assumes that the Dirichlet normalisation factor is constant and proportional to the difference between the Poisson averages $\lambda_t(d, c)$ and $\lambda_t(c)$. This assumption holds for small values of β_t and λ_t, respectively.

BM25-TF-IDF and Poisson Q2D

This section shows a formal proof regarding the conditions that make BM25-TF-IDF a special case of the Poisson log-likelihood ratio of the document.

$$\text{score}_{\text{BM25}}(d, q, c) = \log \frac{P_{\text{Poisson},\lambda}(\vec{k}|q, c)}{P_{\text{Poisson},\lambda}(\vec{k}|c)} \tag{44}$$

For the Poisson model, the standard approach for estimating λ is the product of number of trials and event probability:

$$\lambda_t(x) = n \cdot P(t|x) \tag{45}$$

Where x is a model (e.g. foreground, background, or mixture of the two).
One can find justifications for various settings of n and $P(t|x)$. For BM25-TF-IDF and Poisson, the average occurrence in the background model (collection) may be estimated via the document frequency and the number of documents:

$$\lambda_t(c) = (\text{df}(t, c) + 0.5) \cdot \frac{1}{N_D(c) + 1} \tag{46}$$

Note that $\lambda_t(c) < 1$ since $\text{df}(t, c) < N_D(c)$.
Then, for the average of the posterior model with $\lambda_t(q, c)$, there is a simple setting that explains the BM25-IDF.

$$\lambda_t(q, c) = \begin{cases} 1 - \lambda_t(c) & t \in q \quad n(t, q) > 0 \\ \lambda_t(c) & t \notin q \quad n(t, q) = 0 \end{cases} \tag{47}$$

With this setting, the fraction of averages is:

$$\frac{\lambda_t(q,c)}{\lambda_t(c)} = \frac{N_D(c) - \mathrm{df}(t,c) + 0.5}{\mathrm{df}(t,c) + 0.5} \tag{48}$$

This is the BM25 IDF, eq (4).
Therefore, the Poisson Q2D score, for the described setting of λ_t parameters, is equal to BM25-TF-IDF.

$$\sum_t \mathrm{TF}(t,d,c)\ \log \frac{\lambda_t(q,c)}{\lambda_t(c)} = \mathrm{score}_{\mathrm{BM25}}(d,q,c) \tag{49}$$

Where $\mathrm{TF}(t,d,c)$ is the BM25-TF, eq (3). The sum does not need to be restricted to $t \in q$, since $\lambda_t(q,c) = \lambda_t(c)$ for non-query terms, that is non-query terms do not contribute to the score.

3.2.3 Evaluation

Experimental study

Baselines and Candidate Models

Figure 4 lists the baselines and the candidate models.
For LM (Sec. 3.2.2), there are the two baselines, one with occurrence-based term probability, one with df-based term probability. The BM25-TF-IDF baseline is described in Sec. 3.2.1. Cummins' Dirichlet-multinomial LM is a special case of the Poisson D2Q model.

Parameter Setting P0 The P0 setting is based on *total term count (length)*, and P1 is based on *distinct terms (verbosity)*.
Regarding Dirichlet-multinomial, the P0 setting for the background model:

$$\beta_t(c) = \mathrm{avgdl}(c)\ \frac{n(t,c)}{\sum_i n(t_i,c)} = \frac{n(t,c)}{N_D(c)} = n(t,d_{\mathrm{avg}}) \tag{50}$$

Where $\mathrm{avgdl}(c)$ is the number of trials, $n(t,c)$ is the number of term occurrences, and $N_D(c)$ is the number of documents.
For the foreground model (either document or query), the respective P0 setting is:

$$\beta_t(d) = |d|\ \frac{n(t,d)}{\sum_i n(t_i,d)} = n(t,d) \tag{51}$$

Baselines
LM1: occurrence-based term probability: eq (25) and (26)
LM2: df-based term probability: eq (25) and (27)
BM25-TF-IDF: eq (5)
Cummins' Dirichlet-multinomial LM: eq (29)
equivalent to Poisson D2Q P1
Dirichlet-multinomial D2Q: eq (8)
P0: Length-based
P0: without norm-factor: equivalent to Poisson D2Q P0
P1: Verbosity-based, Cummins
P1: without norm-factor: equivalent to Poisson D2Q P1
Dirichlet-multinomial Q2D: eq (9)
P0: Length-based
P1: Verbosity-based, Cummins
Candidate Models
Poisson D2Q: eq (14)
P0: Length-based
P1: Verbosity-based
Poisson Q2D: eq (15)
P0: Length-based
P1: Verbosity-based

Figure 4: Baseline and candidate models.

Where $|d| = N_L(d)$ is the document length. For the query, the setting is based on $n(t, q)$ and query length, $|q|$.
Regarding the Poisson parameters, the setting is the same:

$$\lambda_t(c) = \varphi \; \beta_t(c) \tag{52}$$

$$\lambda_t(d) = \varphi \; \beta_t(d) \tag{53}$$

Where φ is a scaling parameter, i.e. the total value of the averages does not need to be equal to the Dirichlet parameters.

Parameter Setting P1 The P1 setting is based on *distinct terms*.
For the background model (collection), the setting is:

$$\beta_t(c) = |\overline{d}_{\text{avg}}| \; \frac{n(t, c)}{|c|} \tag{54}$$

Where $\mu_c = |\overline{d}_{\text{avg}}| = N_T(d_{\text{avg}})$ are notations for referring to the number of distinct terms in a document of average length.
For a foreground model (document), the Dirichlet parameter is:

$$\beta_t(d) = |\overline{d}| \; \frac{n(t, d)}{|d|} \tag{55}$$

Where $|\overline{d}| = N_T(d)$ is the number of distinct terms.
For λ_t, the setting is analogous to the Dirichlet case.

$$\lambda_t(c) = \varphi \; \beta_t(c) \tag{56}$$

$$\lambda_t(d) = \varphi \; \beta_t(d) \tag{57}$$

Verbosity: Cummins and Lipani The P1 parameter setting coincides with the notion of verbosity [66]. The verbosity is the number of terms (word count) divided by the number of distinct terms.

$$\text{verb}(d) = \frac{|d|}{|\overline{d}|} \quad \left(= \frac{|d|}{T_d} = \frac{N_L(d)}{N_T(d)} \right) \tag{58}$$

The equation shows the $|d|$ notation, and the notation as in [66] ($\frac{|d|}{T_d} = \frac{N_L(d)}{N_T(d)}$). Then, the verbosity in an average document in collection c is:

$$\text{verb}(d_{\text{avg}}) = \frac{|d_{\text{avg}}|}{\mu_c} = \frac{|d_{\text{avg}}|}{|\bar{d}_{\text{avg}}|} \tag{59}$$

Note that the approaches coincide where μ_c is the number of distinct terms in an average document.

$$\text{verb}(d_{\text{avg}}) = \frac{|d_{\text{avg}}|}{|\bar{d}_{\text{avg}}|} \quad \left(= \frac{|c|}{|T_c|} = \frac{N_L(c)}{N_T(c)} \right) \tag{60}$$

Both, Cummins and Lipani rely on verbosity for estimating parameters, where [66] adds verbosity to the BM25-TF.

Data Sets & Results

Table 1 shows the experimental results for the medical data sets used in this thesis. We report Reciprocal Rank, MAP and DCG for each candidate model, for three benchmarks (OHSUMED, TREC 2004 Genomics track and TREC 2005 Genomics Track). The Poisson D2Q improved all of the measures in OHSUMED and TREC 2004 however, the improvement was not statistically significant. However, plain LM models including LM1 and LM2 received slightly higher values than the Poisson D2Q when applied to TREC 2005. Regarding the genomics datasets, we report smaller MAP than the 0.2171 (The mean of the TREC 2004 MAP values) [107] and 0.1968 (The mean of the TREC 2005 MAP values).

Analysis

The main findings are as follows:

- DCM Q2D work better with the norm factor (OHSUMED, TREC 2004). On the other hand, Poisson D2Q achieved higher performances than DCM D2Q models which implies that the norm factor reduces the effectiveness of the DCM D2Q models.
- Replacing total term frequency with TF-BM25 in Poisson Q2D models was shown to be effective.
- In most of the cases, models with P1 settings outperformed P0 models.

Model	OHSUMED			TREC 2004			TREC 2005		
	R-Rank	**MAP**	**nDCG**	**R-Rank**	**MAP**	**nDCG**	**R-Rank**	**MAP**	**nDCG**
LM1: $n(t,c)/N_L(c)$	0.6631	0.1441	0.2906	0.6265	0.1299	0.1879	0.5456	0.1088	0.2095
LM2: $\mathrm{df}(t,c)/\sum_t \mathrm{df}(t,c)$	0.6739	0.1475	0.2947	0.6419	0.1303	0.1885	0.5405	**0.1107**	**0.2108**
BM25-TF-IDF	0.6791	0.1454	0.2947	0.6468	0.1287	0.1872	0.5306	0.0991	0.2012
Dirichlet-multinomial									
Q2D: P0	0.4879	0.1025	0.2432	0.4777	0.1001	0.1677	0.3749	0.0736	0.1682
Q2D: P0: no-norm	0.4695	0.0960	0.2346	0.4504	0.0970	0.1628	0.3779	0.0739	0.1681
Q2D: P1	0.4923	0.1012	0.2418	0.4679	0.0991	0.1673	0.3930	0.0734	0.1687
Q2D: P1: no-norm	0.4344	0.0923	0.2279	0.4777	0.0928	0.1613	0.2881	0.0553	0.1464
D2Q: P0	0.5266	0.1093	0.2504	0.4510	0.0944	0.1641	0.3636	0.0667	0.1624
D2Q: P0: no-norm	equivalent to Poisson D2Q P0								
D2Q: P1	0.5359	0.1096	0.2515	0.4628	0.0977	0.1680	0.3635	0.0698	0.1640
D2Q: P1: no-norm	equivalent to Poisson D2Q P1								
Poisson									
Q2D: P0	0.4783	0.1216	0.2568	0.4694	0.1155	0.1728	0.4354	0.0830	0.1782
Q2D: P0: BM25-TF	0.5817	0.1324	0.2741	0.5373	0.1188	0.1785	0.4342	0.0917	0.1850
Q2D: P1	0.5066	0.1222	0.2583	0.4965	0.1159	0.1740	0.4338	0.0823	0.1785
Q2D: P1: BM25-TF	0.6106	0.1356	0.2794	0.5476	0.1193	0.1795	0.4500	0.0928	0.1870
D2Q: P0	0.7215	0.1444	0.2918	0.6574	0.1293	0.1892	**0.5610**	0.0974	0.2021
D2Q: P1	**0.7313**	**0.1531**	**0.3005**	**0.6932**	**0.1325**	**0.1917**	0.5584	0.1063	0.2092

Table 1: Experimental results: Overall, Poisson D2Q (with parameter setting P1) is the best performing model, with baseline LM2 being superior for MAP and nDCG for TREC-5. The strong effect of using BM25-TF instead of the total TF count is evident for Ohsumed and TREC-4.

3.2.4 Discussion

Dirichlet-multinomial, Poisson and BM25

Dirichlet-multinomial LM (D2Q) as a ranking score is based on the Dirichlet-multinomial distribution, and thus comes with a strong theoretical justification. Regarding the BM25-TF-IDF, theoretical justifications remain an issue, even though the probability of relevance framework, the 2-Poisson model, and divergence of randomness can be employed for creating justifications. The Poisson model comes as a convenient and concise model to serve as an explanation for query-generating Dirichlet-multinomial and as an explanation for BM25-TF-IDF.
While the retrieval quality for Poisson D2Q was expected to be high (since it is equivalent to Cummins' Dirichlet-multinomial approach), one of the main outcomes of this section was to measure and clarify the D2Q versus the Q2D approaches.

Query-generating (D2Q) vs Document-generating (Q2D)

TF Quantification While for D2Q, where the query is the target, the document plays the role of the foreground model, and the term frequency $n(t, d)$ is reflected in the foreground parameters ($P(t|d)$ for multinomial, $\beta_t(d)$ for Dirichlet, and $\lambda_t(d)$ for Poisson), for Q2D, where the document is the target, one needs a saturating TF quantification $\mathrm{TF}(t, d, c)$ to replace the total term count $n(t, d)$. The strong effect is evident of Ohsumed and TREC-4, for Poisson Q2D candidates. Of course, this effect is well understood [88, 90], but until today the difference between D2Q and Q2D and total vs saturated count is not common knowledge. The way Cummins in [23] applies DCM confirms that the D2Q-side of retrieval can be confusing regarding the effect of TF saturation. However, it was important to reconfirm the effect for the Poisson-based formulation of the respective scores.

Non-Query Term Assumption For a query-generating model, non-query terms drop out because of the way the term frequency $n(t, q)$ is considered. For query terms not in the document, a penalty is applied.
For a document-generating model, non-document terms drop out. For document terms not in the query, a penalty *would be* applied, if there were no non-query term assumption. This is problematic, since there are many document terms, and the penalty overpowers the effect of matching terms.

Therefore, for Q2D, a non-query term assumption is required. Mathematically, this can be solved by considering the product of TF's $\mathrm{TF}(t,d,c)\ \mathrm{TF}(t,q,c)$ or by restricting the sum over terms to the sum over query terms, $\sum_{t \in q}$. Both of these approaches essentially express that the mixture of foreground and background is equal to the background parameter for the case of non-query terms. For example:

$$\lambda_t(q,c) = \lambda_t(c) \qquad t \notin q$$

Such consideration is not new, is not specific to the proposed Poisson-based formulation of retrieval scores. LM, as a query-generating model, will apply a penalty for non-query terms, while BM25, as a document-generating model, will nullify the effect of non-query terms ($\mathrm{TF}(t,q,c) = 0$ for non-query terms).

Summary

The main outcome of this section is the formulation of retrieval scores via the Poisson model (Sec. 3.2.2). The two scores defined (def 3, eq (14) and def 4, eq (15)) were:

$$\mathrm{score}_{\text{Poisson-Q2D}}(d,q,c) := \sum_t \mathrm{TF}(t,d,c)\ \log \frac{\lambda_t(q,c)}{\lambda_t(c)}$$

$$\mathrm{score}_{\text{Poisson-D2Q}}(d,q,c) := \sum_t \mathrm{TF}(t,q,c)\ \log \frac{\lambda_t(d,c)}{\lambda_t(c)}$$

While for D2Q, the penalty induced by $\lambda_t(d,c) < \lambda_t(c)$ is acceptable for the score, for Q2D, the penalty arising from $\lambda_t(q,c) < \lambda_t(c)$ for the many document terms that are not in the query, tends to overpower the effect of matching terms. Therefore, the non-query-term assumption with $\lambda_t(q,c) = \lambda_t(c)$ will be applied.

The Poisson scores extend the family of major baseline models. Currently, multinomial LM, Dirichlet-multinomial, and BM25-TF-IDF are dominating baselines. In future, the Poisson scores can be considered as standard retrieval scores, while some of the existing models are special cases of the Poisson scores.

For this section, we left aside the DFR branch of models. Obviously, the Poisson model is a first choice for a model of randomness, and the log-likelihood ratio measures the divergence between posterior and prior model. A study

regarding the proposed Poisson IR model and DFR is subject of a future publication.
Regarding the focus on Dirichlet-multinomial and BM25-TF-IDF, there are two formal proofs (Sec. 3.2.2) showing under which conditions the Dirichlet-multinomial D2Q approach reduces to the Poisson D2Q score, and BM25-TF-IDF is equivalent to the Poisson Q2D score. The Poisson-based formulation of retrieval scores is intuitive and concise, and the scores deliver the same performance as their more complicated siblings.
Overall, this section establishes Poisson-based scores as an easier way of formulating query-generating Dirichlet-multinomial LM and document-generating BM25. Interestingly, this research brings back the Poisson model, the model that led to the BM25-TF [85], has been used for explaining the IDF [71], but then disappeared between BM25, multinomial and Dirichlet-multinomial, and DFR.
Through the concise formulation of Poisson-based retrieval scores, IR research and other disciplines gain new methods that perform well and are closely related to existing scores. Such mathematically grounded formulations of retrieval scores can be expected to be considered as alternatives to divergence-related measures (e.g. point-wise mutual information in NLP) or geometric similarity scores (e.g. cosine similarity between test and training candidates). Thereby, the parameter estimation for the Poisson parameter λ is guided by what [28] established for Dirichlet-multinomial classification/clustering, [116] established for Dirichlet-multinomial retrieval, [23] established for Dirichlet-multinomial LM. The main difference is that the Poisson-based formulation is lighter than the Dirichlet-multinomial one, and that there is a strong connection between Poisson and BM25.

Chapter 4

Semantic IR

4.1 Chapter overview

In section 4.2, we review the basis of a generalizable knowledge-oriented ranking framework built upon the dimensions of the collection including classes and terms. Moreover, we perform a naive analysis on query complexity in the medical domain. The primary focus of this thesis is not to investigate query complexity, however a deep analysis of this factor and its relationship with IR is a good research direction for future work (because concepts such as semantic complexity could help in deciding suitable IR models for given queries). We briefly explore the influence of information needs on forecasting the retrieval quality and subsequently categorize genomics topics based on semantics. We employ the well-established bag of words (BOW) models and their semantic extensions to perform an individual query analysis. Within the analysis process, we compare the quality of plain Dirichlet-multinomial language modelling with its semantically reinforced model to study the role of entity burstiness in knowledge retrieval. We measure the quality of the models across a range of different query types discriminated by the query intent and the structure. The analysis task builds the formal grounds for the development of a decision-making framework which determines to what extent models should be semantic-aware.
One of the substantial contributions of this thesis is to show how to aggregate scores of semantic models leading to a reliable combined model. We considered the use of Query Performance Prediction (QPP) for this task. Although a specific IR model might be shown to be effective when applied

on a medical collection like Medline, there is no way to infer that this model is certainly effective when moved to other collections of the same task (such as health-related notifications). The existence of different representations of clinical information needs is a key parameter that impacts the success rate of IR models [57]. The increasing diversity in the performance of the representations of the information needs led to a new research direction; namely, Query Performance Prediction (QPP) or Query Difficulty Estimation (QDE) [20]. Studying QPP in section 4.3 is important since we use QPP for the development of the integration parameters concerning combing RSV (Retrieval Score Value) scores of the semantic instances of FDCM in eq (102)(page 81) and RFDCM in eq (146)(page 115). The well-known pre-retrieval predictors such as Average Inverse Document Frequency (AvgIDF), Average Inverse Collection Term Frequency (AvICTF) and Simplified Clarity Score (SCS) are derived from the statistical features of the queries and in section 4.3 we aim to deeper study word burstiness for developing novel predictors.
Within the categorization of medical entities, we often come across various terms for the same concept [1]. Intuitively, if a document starts with a term in relation to a concept and the author intends to repeat the concept, it is more likely that they will continue to reuse that specific term. This phenomenon is a type of term dependency which is known as word burstiness [23, 28]. The multinomial probability distribution is a common approach to model the documents but it does not account for word burstiness [28]. Many applications apply the heuristics by topping IR models off with some novel parameters to deliver burstiness identification into the retrieval process. However, these heuristics are not generalizable, and their theatrical explanations are rarely published [89].
In section 4.3, we propose a set of novel pre-retrieval predictors that are based on burstiness identification (DCM background model and natural harmony assumption), position-based term probability and the amount of information carried by the query terms. Later in section 6.2.2, we use the proposed natural harmony predictor as a feature in training a IR-based recommender system. These predictors could be later used for the development of a probabilistic framework that predicts the optimal IR models for the given queries against the health-related notifications.
Medline is a commonly used benchmark for searching the biomedical literature. It is maintained by National Library of Medicine (NLM) and as March

2018, it contained more than 24 million references to journal articles [1]. We conducted some experiments on the Medline citations and a collection of 25 queries used in 2012, 2011 and 2007 Text Retrieval Conference (TREC) Medical and Genomics tracks to capture the correlations between the novel features and the well-established predictors.

The main contribution of this chapter is proposing a combined semantic model (terms+concepts) in section 4.4. Concept-based IR is expected to improve the quality of medical ranking since it captures more semantics than BOW representations. However, bringing concepts and BOW together into a transparent IR framework is challenging. Therefore, in section 4.4, we propose a new aggregation parameter to combine conceptual and term-based DCM scores. The determination of this linear parameter is the result of exploring to what degree the difference of the conceptual and term-based sum of IDFs is influential to the integration. Instead of employing machine learning or heuristics to find combined models, this section aims to establish reasonable aggregation standards based on semantic query performance predictors (QPPS). The aim of introducing such standards is not to prove that they could or could not outperform advanced ML (e.g. recent works in Neural Networks) but to infer that these models are reliable answers to many research questions and should not be forgotten due to their flexibility, understandability and explainability nature. This approach is a starting point for tackling the lack of transparent conceptual IR standards for urgent notification filtering.

The section aims to achieve clear and light-weighted theoretical foundation and standardisation through the extension of the ability of IR to capture semantics. By establishing standards, we are enabling data scientists to use the technology in the health sector and to easily incorporate the output of the different information extraction tools into the process. This section helps data scientists to explore the semantic dimensions of the query and the documents to apply advanced probabilistic retrieval models on health-related applications.

[1] https://www.nlm.nih.gov/pubs/factsheets/medline.htmll

4.2 Preliminary semantic IR and query complexity

4.2.1 Background

The use of semantics including concepts and relations in IR has been well established, for example [6] represented semantic instances of TF-IDF which are easily extendable to other traditional IR models. This section intends to provide a general overview of the preliminary semantic IR instances and later in the contribution briefly discuss the importance of semantic complexity in respect to the quality of IR. The semantic complexity is the number of "things" in a given query including objects, the attributes derived from the objects and the relationships between the objects that are expressible by the users [83]. The background section also shows the relational content model for representing semantics.

Semantic knowledge representation

Table 2 is the representation of the probabilistic object relational content model of the example query "Hypercarbia effects Pathogenesis of hemorrhage", which shows the mappings between terms, medical concepts (classes) and relationships (subject, object and verb). The term-doc table stores the terms of Medline citations along with their relevant PubMed Unique Identifiers (PMIDs). Additionally, the relationships table is used to index the text-based contents based on a data structure constructed from subject, verb and object. A sample example of the model is represented in table 2.

Term	**PMID**
Hypercarbia	30963
Pathogenesis	30963
Hemorrhage	30963

ClassName	**Object**	**PMID**
Condition	Hypercarbia	30963
Biological Mechanism	Pathogenesis	30963
Event	Hemorrhage	30963

The probabilistic object relational content model enables the semantic models to leverage from factual knowledge which is shown in figure 5. As can

Relationship	**Subject**	**Object**	**PMID**
effect	Hypercarbia	Pathogenesis of hemorrhage	30963

Table 2: Probabilistic object relational content model representing a medical phrase.

be seen, the article is transformed into a classification file containing a list of medical entities and their descriptions within the document. The *Object* field stores a term text mapped to a concept while the entity-type-hierarchy is represented in the *class* and *entity* fields. The class frequency of the document is represented in the *freq* attribute of the *Object* field. We process the *Object* fields to match the query terms with medical entities and accordingly extract relevant class frequencies.

```
<Classifications>
    <class name="disorder">
       <entity name="Infectious canine hepatitis"/>
        <Object freq="5"> ich </Object>
    </class>
    <class name="body structure">
        <entity name="Entire lung">
        <Object freq="12"> lung </Object>
    </class>
    <class name="regime/therapy">
        <entity name="Providing presence"/>
        <Object freq="1"> presence </Object>
    </class>
    <class name="substance">
    <entity name="Monosodium glutamate"/>
        <Object freq="2"> msg </Object>
    </class>
    <class name="finding">
        <entity name="Nursing diagnosis"/>
        <Object freq="1"> nd </Object>
    </class>
</Classifications>
```

Figure 5: Knowledge representation (XML).

Knowledge-oriented retrieval models

We transfer term-based IR models to their corresponding semantic extensions. The framework is able to consolidate the models derived from terms, classifications, relationships and attributes (and any other semantic dimension) into the macro and micro combined models, however the focus of this section is the mixture of terms and classifications.

XF.IDF This is a combination of TF-IDF (term-oriented model) and semantic extensions such as CF-IDF (Classification-based TF-IDF). The TF-IDF weight is the result of multiplying TF and IDF scores. Some well-known IDF variants are listed in definition (5)(page 64).

$$\mathrm{W}_{\text{TF-IDF}}(t,d,q,c) := \mathrm{TF}(t,q) \cdot \mathrm{TF}(t,d) \cdot \mathrm{IDF}(t,c) \tag{61}$$

The Retrieval Status Value (RSV) is the sum of TF-IDF weights across the query vector.

$$\mathrm{RSV}_{\text{TF-IDF}}(d,q,c) := \sum_{t} \mathrm{W}_{\text{TF-IDF}}(t,d,q,c) \tag{62}$$

$\mathrm{TF}(t,d)$ is the within-document term frequency and $\mathrm{IDF}(t)$ is the Inverse Document Frequency component. One popular variant of IDF is the negative logarithm of the document frequency in proportion to the total number of documents. The semantic micro instance of TF-IDF is as follows:

$$\mathrm{RSV}_{\text{XF.IDF}}(d,q,c) := \sum_{x} \mathrm{XF}(x,q) \cdot \mathrm{XF}(x,d) \cdot \mathrm{IDF}(x,c) \tag{63}$$

To make the formulations readable, we used type aware x functions where XF is the frequency of semantic type x. $\mathrm{TF}(t)$ is the term frequency of term t in the document and $\mathrm{CF}(c)$ is the frequency of class c in the document. We proposed the below semantic aware macro XF.IDF built upon the linear mixture of the scores derived from different semantic dimensions:

$$\mathrm{RSV}_{\text{XF.IDF-macro}}(d,q,c) := \sum_{x \in \{T,C,R,A\}} s_x \cdot \mathrm{RSV}_{\text{XF.IDF}}(d,q,c) \tag{64}$$

where T is terms, C is classes, R is relationships and A is attributes. s_x is the linear mixture parameter for each predicate.

XLM Eq (65) shows the term-based language modelling (TLM).

$$\mathrm{RSV}_{\mathrm{TLM}}(d,q,c) := \sum_{t} \mathrm{W}_{\mathrm{TLM}}(t,d,q,c) \tag{65}$$

$$\mathrm{RSV}_{\mathrm{XLM}}(d,q,c) := \sum_{x} \mathrm{XF}(x,q) \cdot \log\left(\frac{(1-\sigma_d)\cdot p(x|c) + \sigma_d \cdot p(x|d)}{p(x|c)}\right) \tag{66}$$

$$\mathrm{RSV}_{\text{XLM-macro}}(d,q,c) := \sum_{x\in\{T,C,R,A\}} s_x \cdot \mathrm{RSV}_{\mathrm{XLM}}(d,q,c) \tag{67}$$

XDCM The term-based DCM is comprehensively discussed in section 4.4.1. Let x, be a semantic dimension, $\alpha_c(x)$ be the background model, $\alpha_d(x)$ be the document model and XF be the frequency of x, XDCM is defined as follows:

$$\mathrm{RSV}_{\mathrm{XDCM}}(d,q,c) := \sum_{x} \mathrm{XF}(x,q) \cdot \log\left(\frac{(1-\sigma_d)\cdot \alpha_c(x) + \sigma_d \cdot \alpha_d(x)}{\alpha_c(x)}\right) \tag{68}$$

$$\mathrm{RSV}_{\mathrm{XDCM-macro}}(d,q,c) := \sum_{x\in\{T,C,R,A\}} s_x \cdot \mathrm{RSV}_{\mathrm{XDCM}}(d,q,c) \tag{69}$$

The aggregation parameter s_x is in range of the interval (0-1). Tuning this parameter for (terms+concepts) is discussed in section 4.4.2 and for (items+concepts in recommendation) is discussed in section 6.2.2.

4.2.2 Contribution

Query classification

Table 3 shows how I manually classified the topics of TREC-2004 and TREC-2005 genomics data into five categories based on a consistent pattern I found

in their information needs. M queries intend to retrieve plain lists of medical concepts and PM queries intend to retrieve sets of properties linked to some specific medical concepts. Explorative-Item (EI) queries are supposed to retrieve actual documents, articles, reports or properties related to Items e.g. 'parts of a document' or 'focus of a study'.

Dataset	**Acronym**	**Description**
TREC-2004	M	Retrieval of Medical Concepts.
TREC-2004	PM	Properties of a specific Medical Concept e.g. Function of a protein or information/research related to a Concept.
TREC-2004	EI	Explorative-Item retrieval. The need is Documents/Articles/Reports/studies or properties related to some items (e.g. 'focus of studies').
TREC-2005	PI	The need is providing information concerning a specific Medical Entity.
TREC-2005	DP	Queries intend to retrieve a description of some methods/procedures that are associated with a Medical Entity.

Table 3: Descriptions of the information need types.

Moreover, I defined information needs associated with topics used in TREC-2005. The need of the PI class is information about the medical concepts whereas, DP queries seek descriptions of methods and procedures associated with the medical concept.

I demonstrate the topics with their semantic analysis in table A.1 and table A.2 of appendix A. This analysis consists of query length, query language (declarative or interrogative) and the classifications of information needs.

4.2.3 Evaluation

Experimental setup

The named entity recognition task filters out the noun phrases within the row texts by combining *NLTK* and *Spacy* libraries. An extracted noun phrase is either a single term or a compound phrase (mixture of adjectives and noun tokens). We employ the Systematized Nomenclature of Medicine Clinical Terms (SNOMED CT) to match the medical entities from the list of the noun phrases.
We utilized *wordnet* to match up the terms with the relevant medical concepts and their synonyms. *Wordnet* is an online lexical database. It groups English terms into lists of synonyms called synsets [75]. Moreover, syntactic dependency parser as part of the *Spacy* was used to identify subject, verb and object triples. I considered plural to singular conversion as well in order to improve the effectiveness of the semantic knowledge representation.
We converted the query string into a list of noun phrases and mapped the results to the corresponding medical entities by leveraging the SNOMED CT dictionary. This process led to the extraction of a hierarchy-shaped list containing of entity types. The *Spacy* dependency module was used for parsing. This process infers whether a part of speech is subject/object or verb. We generated a list of subject–verb–object (SVO) triples corresponding to sentences within the queries. *wordnet* was used to compute the synonyms for the verbs and embedded them into the query formulation schema. Figure 6 shows an example of a formulated query.
We show the individual query analysis as well as the topic quality evaluation in table 4 and table 5.

Evaluation

Analysis based on query intent

I compared the retrieval quality of the semantic approach against query intent and complexity by experimenting on 65 genomics topics used in 2004 and 2005 Text Retrieval Conference (TREC). We limited the query set to 60

QN	QT	XDCM	XLM	TDCM	TLM	Avg (X)	Avg (T)	diff (X-T)
25	PM	0.7857	0.7914	0.7147	0.7302	0.788 55	0.722 45	0.0661
27	M	0.064	0.0663	0.0368	0.046	0.065 15	0.0414	0.023 75
29	M	0.1064	0.1138	0.0951	0.1033	0.1101	0.0992	0.0109
6	M	0.0344	0.038	0.0253	0.0256	0.0362	0.025 45	0.010 75
3	M	0.0144	0.0347	0.0118	0.018	0.024 55	0.0149	0.009 65
12	PM	0.0462	0.0443	0.0391	0.0344	0.045 25	0.036 75	0.0085
18	EI	0.1885	0.1896	0.1865	0.179	0.189 05	0.182 75	0.0063
35	PM	0.0191	0.019	0.0154	0.0128	0.019 05	0.0141	0.004 95
30	M	0.0311	0.0309	0.0299	0.0298	0.031	0.029 85	0.001 15
32	Y/NO	0.01	0.0098	0.0088	0.0088	0.0099	0.0088	0.0011
11	EI	0.0012	0.0015	0.0003	0.0003	0.001 35	0.0003	0.001 05
22	PM	0.1127	0.1124	0.1121	0.1128	0.112 55	0.112 45	0.0001
23	EI	0.0003	0.0003	0.0002	0.0002	0.0003	0.0002	0.0001
9	EI	0.0327	0.0303	0.0334	0.0304	0.0315	0.0319	−0.0004
10	EI	0.0025	0.0026	0.0036	0.0038	0.002 55	0.0037	−0.001 15
28	M	0.0228	0.0219	0.0252	0.022	0.022 35	0.0236	−0.001 25
2	PM	0.0479	0.0487	0.0473	0.0541	0.0483	0.0507	−0.0024
20	EI	0.0332	0.0336	0.0359	0.0364	0.0334	0.036 15	−0.002 75
13	PM	0.0079	0.0074	0.0108	0.0108	0.007 65	0.0108	−0.003 15
26	PM	0.1557	0.1588	0.1586	0.1669	0.157 25	0.162 75	−0.0055
4	EI	0.0075	0.0098	0.0128	0.0175	0.008 65	0.015 15	−0.0065
21	EI	0.0498	0.0485	0.0574	0.0573	0.049 15	0.057 35	−0.0082
1	EI	0.0871	0.0743	0.0949	0.0861	0.0807	0.0905	−0.0098
17	EI	0.0359	0.0318	0.0477	0.0444	0.033 85	0.046 05	−0.0122
31	EI	0.1139	0.1221	0.1442	0.1458	0.118	0.145	−0.027
15	EI	0.0762	0.0879	0.106	0.1193	0.082 05	0.112 65	−0.0306
33	EI	0.066	0.0631	0.0997	0.0935	0.064 55	0.0966	−0.032 05
16	PM	0.4714	0.4528	0.5051	0.4992	0.4621	0.502 15	−0.040 05
19	EI	0.1031	0.0794	0.1615	0.1348	0.091 25	0.148 15	−0.0569
14	Y/NO	0.1532	0.1529	0.2141	0.2293	0.153 05	0.2217	−0.068 65
24	PM	0.3745	0.3778	0.4542	0.4564	0.376 15	0.4553	−0.079 15
5	EI	0.2747	0.2652	0.3535	0.3564	0.269 95	0.354 95	−0.085
34	EI	0.0963	0.0716	0.2495	0.2033	0.083 95	0.2264	−0.142 45
7	EI	0.2633	0.244	0.4453	0.443	0.253 65	0.444 15	−0.1905
8	EI	0.0722	0.1341	0.5203	0.5117	0.103 15	0.516	−0.412 85

Table 4: TREC-2004 MAP analysis of queries. Avg (X) is the average MAP value of XLM and XDCM. Avg (T) is the average MAP value of TLM and TDCM. The last column lists differences between Avg (X) and Avg (T).

QN	QT	XDCM	TDCM	XLM	TLM	Avg (X)	Avg (T)	diff (X - T)
17	PI	0.2401	0.1778	0.2386	0.1537	0.239 35	0.165 75	0.0736
11	PI	0.0278	0.0126	0.0309	0.0135	0.029 35	0.013 05	0.0163
5	DP	0.0179	0.0066	0.0227	0.0069	0.0203	0.006 75	0.013 55
21	PI	0.0423	0.0297	0.0443	0.0301	0.0433	0.0299	0.0134
7	DP	0.025	0.02	0.0308	0.0252	0.0279	0.0226	0.0053
29	PI	0.1454	0.1402	0.1463	0.1415	0.145 85	0.140 85	0.005
19	PI	0.2761	0.2723	0.2796	0.2745	0.277 85	0.2734	0.004 45
16	PI	0.1215	0.1185	0.128	0.1229	0.124 75	0.1207	0.004 05
22	PI	0.0399	0.0364	0.0418	0.0378	0.040 85	0.0371	0.003 75
26	PI	0.0037	0.0024	0.0037	0.0025	0.0037	0.002 45	0.001 25
13	PI	0.1784	0.177	0.1798	0.179	0.1791	0.178	0.0011
24	PI	0.0074	0.0071	0.0076	0.0073	0.0075	0.0072	0.0003
14	PI	0	0	0	0	0	0	0
3	DP	0.0007	0.0012	0.0008	0.0013	0.000 75	0.001 25	−0.0005
23	PI	0.005	0.0055	0.0049	0.0055	0.004 95	0.0055	−0.000 55
30	PI	0.0861	0.085	0.1126	0.1161	0.099 35	0.100 55	−0.0012
15	PI	0.0005	0.0035	0.0004	0.002	0.000 45	0.002 75	−0.0023
12	PI	0.007	0.0103	0.0064	0.0086	0.0067	0.009 45	−0.002 75
4	DP	0.0082	0.0161	0.0068	0.0103	0.0075	0.0132	−0.0057
27	PI	0.0233	0.0296	0.0236	0.0295	0.023 45	0.029 55	−0.0061
1	DP	0.0119	0.019	0.013	0.0211	0.012 45	0.020 05	−0.0076
9	DP	0.0223	0.0308	0.023	0.0305	0.022 65	0.030 65	−0.008
6	DP	0.0284	0.053	0.0288	0.0601	0.0286	0.056 55	−0.027 95
20	PI	0.0346	0.0655	0.0368	0.0666	0.0357	0.066 05	−0.030 35
10	DP	0.2736	0.3174	0.2682	0.3032	0.2709	0.3103	−0.0394
25	PI	0.0281	0.0743	0.0333	0.0836	0.0307	0.078 95	−0.048 25
2	DP	0.0352	0.0976	0.0381	0.1073	0.036 65	0.102 45	−0.0658
8	DP	0.1038	0.1707	0.1046	0.1755	0.1042	0.1731	−0.0689
18	PI	0.2973	0.4038	0.2893	0.3569	0.2933	0.380 35	−0.087 05
28	PI	0.3686	0.4504	0.3872	0.5173	0.3779	0.483 85	−0.105 95

Table 5: TREC-2005 MAP analysis of queries. Avg (X) is the average MAP value of XLM and XDCM. Avg (T) is the average MAP value of TLM and TDCM. The last column lists differences between Avg (X) and Avg (T).

```
Terms = [Paracetamol, can, cure, flu]
Entities = [Paracetamol, flu]
Ontology (PORCM):{
    instanceOf( Flu, Disorder ),
    is_a ( Paracetamol, Painkiller),
    is_a (Painkiller, Drug)
                }
    Relationship-Triples = [(Paracetamol,
cure[heal,threat],Flu),...]
```

Figure 6: Knowledge-based query representation.

topics by filtering the queries in which the named entity recognizer was able to detect the medical concepts. The Topics have been discriminated by length and the level of semantic complexity (type of information need).
I measured the average MAP (Avg-MAP) scores of the individual topics for both term-only (TLM in eq (65) and TDCM in eq (29)) and semantic (XLM in eq (66) and XDCM in eq (68)) baselines. The last columns of table 4 and table 5 listed the differences of the Avg-MAP values of semantic and term-only approaches (columns are sorted in descending order). Since there were small numbers of topics in some groups, comparing baselines by the use of statistical significance tests would not be very reliable for quality determination and therefore, I divided the number of queries where the difference (Semantic - BOW) > 0 (for semantic approach quality check) and where the difference < 0 (for the BOW approach quality check) by the total number of the queries classified in that group. This analysis can be seen in figure 7.
Concerning M queries, the semantic models gave rise to the importance of the medical concepts and as a result they were more effective than BOW models. On the other hand, the BOW approach provided a significantly higher quality when applied to explorative queries.
As the tables show, the quality of the BOW models compared to the semantic models is more acceptable and consistent across DP queries. This is due to the disability of the semantic models in giving rise to the non-medical phrase "*procedures and methods*" which play an important role in the determination of the information need.
If we assign more weights to the words which come after the interrogative keywords such as *what* and *where* or informative words (e.g. *Find* and *describe*), the quality would go dramatically higher in respect to fact-based topics. PI queries also worked better with the semantic approach.
The semantic approach was ineffective when the indicator of the information

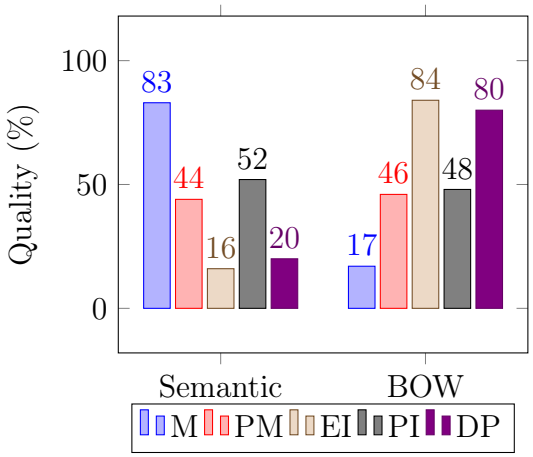


Figure 7: Retrieval quality for different information needs: The semantic models were more effective for M and PI queries compared to BOW models. The descriptions of the query types used in the experiments are shown in table 3.

need is a compound phrase constructed from more than one term. For example *(adenomatous polyposis coli) gene* is a complex compound concept within query "*Provide information on the role of APC (adenomatous polyposis coli) gene in the process of the assembly of the actin*".
In addition to concept mapping failure, if a topic is constructed from many medical concepts, the semantic models might not be able to detect the need of the topic which results in lower performance. Therefore, it could be interesting to consider the complexity of the topic structure and develop a more sophisticated and intelligent tool to analyze the needs of complex queries. Furthermore, the quality of the semantic models is variable regarding queries of different length. This finding inspired us to use query length for developing the FDCM term-concept aggregation parameter in section 4.4.2.
The collection frequencies of compound medical concepts embedded in queries are low which leads to giving unnecessary rise to parts of queries and decrease the quality. Consequently, future studies could look at tuning the collection frequency problem of semantic models.

4.2.4 Discussion

We have explored a preliminary family of semantic retrieval models based on terms and classifications. The transformation from the traditional keyword-based retrieval is a result of employing a schema that represents the content and factual knowledge in one compliant framework.
We investigated the quality of the semantic approach against query intent and length (complexity). This analysis was performed by comparing the average MAP values of individual queries for each classification of information needs against term-only (average MAP of TLM in eq (65) and TDCM in eq (29)) and semantic (average MAP of XLM in eq (66) and XDCM in eq (68)) scores. The experimental results revealed that the effectiveness of the semantic models is reliable and consistent against fact-based queries. However, the BOW models provided a higher quality for the explorative queries. This study confirmed that the semantic-aware models fit well with fact-based queries. For this section, we applied a semantic representation limited to the classification of entities. Future studies will explore the use of relationships and attributes.

4.3 Query Performance Predictors (QPPs)

4.3.1 Background

Preliminaries

Query performance predictors (QPPs) are statistical features for estimating the effectiveness of a search (IR model) performed in response to a query with no relevance judgments [117]. Effectiveness of QPPs has been studied in many papers. He and Ounis [43] measured the linear correlations of six pre-retrieval predictors with average precision. Their experimental results showed that the Simplified Clarity Score (SCS) and the Average Inverse Collection Term Frequency (AvICTF) perform better compared to other candidates. Furthermore, Sondak et al. [97] proposed a QPP framework that gives rise to the effectiveness of the query representation. Moreover, Carmel and Yom-Tov [20] discussed the recent parameters that influence the query difficulty estimation. They compared the correlations between the following pre-retrieval predictors:

AvgIDF In 1972 Karen Spärck Jones proposed a measure to compute the term weights which later became known as Inverse Document Frequency (IDF) [86]. The intuition was that a term which is observed in a multitude number of documents is not a suitable discriminator and this led to the heuristic implementation of IDF. The IDF weight especially when tied up with TF, is an essential leap in IR. Previous studies demonstrated that smoothed language modelling (LM) is correlated to IDF and even TF-IDF can be justified as a normalized version of LM. The relation between TF-IDF and LM has been studied in [88, 4].
The IDF equation is correlated to one version of Zipf's law which states that if we plot a graph of the log of frequency against the log of rank, the outcome will be a straight line [86]. Below we listed some well-known variants of IDF.

Definition .5 (IDF variants)

$$\text{IDF}(t) =: \log\left(\frac{N}{df_t}\right) = -\log\left(\frac{df_t}{N}\right) \tag{70}$$

$$\text{IDF}_{smooth}(t) =: \log\left(\frac{N}{1+df_t}\right) + 1 \tag{71}$$

$$\text{IDF}_{probabilistic}(t) =: \log\left(\frac{N-df_t}{df_t}\right) \tag{72}$$

Some pre-retrieval predictors were derived from the IDF quantification such as SumIDF, AvgIDF and MaxIDF. Scholer et al. [93] conducted experiments to predict the query performance based on IDF. They showed that the maximum IDF (MaxIDF) of the query terms provides the highest correlations on TREC web data. Also, the effectiveness of MaxIDF is discussed in [121].

$$\text{SumIDF}\,(q) = \sum_{t \in q} \log(\frac{N}{df_t}) \tag{73}$$

$$\text{AvgIDF}\,(q) = \frac{\text{SumIDF}(q)}{|q|} \tag{74}$$

In equations above, df_t denotes the frequency of documents in which the term t is observed.

SCS Simplified Clarity Score is essentially the relative entropy or Kullback-Leibler (KL) divergence between the query and collection unigram language models [22]. This pre-retrieval predictor has a considerable impact on the performance due to its intrinsic role in the estimation of the query clarity [43].

$$\mathrm{D}_{KL}(q) = SCS(q) = \sum_{t \in q} p(t|q) \cdot \log_2 \frac{p(t|q)}{p(t|c)} \tag{75}$$

SCQ The Collection Query Similarity measures the similarity between the query and the collection.

$$\mathrm{SCQ}(t) = (1 + \log(n(t,c))) \cdot IDF(t) \tag{76}$$

$$\mathrm{MaxSCQ}(q) = \max_{t \in q} \mathrm{SCQ}(t) \tag{77}$$

AvgPMI Pointwise Mutual Information is a feature based on the co-occurrence statistics of the query terms. AvgPMI is the average of all PMI scores across possible pairs that can be constructed from the query terms. Accordingly, a high AvgPMI indicates that the query terms are strongly correlated.

MaxVAR Maximum variance of the query term weights distribution.

4.3.2 Contribution

proposed pre-retrieval predictors

We propose a set of novel pre-retrieval predictors based on word burstiness, position-based term probability and the amount of information carried by the query terms. The novel predictors are AvgTF, PosTF-IDF, DCBackgroundModel and NaturalHarmony.

AvgTF This quantification denotes the Average Term Frequency over *term-elite* documents [88]. A *term-elite* document is any document in which the current term is observed. The consideration of *term-elite* only documents is for capturing the average frequency/rareness of a term in a document that

includes the term. Lets $n(t,c)$ be the term frequency of term t in collection c, AvgTF and SumAvgTF are defined as follows:

$$\text{AvgTF}(t) = \frac{n(t,c)}{df_t} \tag{78}$$

$$\text{SumAvgTF}(q) = \sum_{t \in q} \text{AvgTF}(t) \tag{79}$$

PosTF-IDF $TF(t,q)$ is the traditional within query term frequency. We tune the within query term frequency based on the position of each term in the query, as intuitively the first and last words in a query sequence carry more information.

$$\text{PosTF}(t,q) = \begin{cases} n(t,q) + 2, & \text{if } position = 0 \\ n(t,q) + 1, & \text{if } position = n-1 \\ n(t,q), & \text{otherwise} \end{cases} \tag{80}$$

$$\text{PosTF-IDF}(q) = \sum_{t \in q} \text{PosTF}(t) \cdot \text{IDF}(t) \tag{81}$$

NaturalHarmony Roelleke et al. [89] proposed some instances of the assumption functions based on a generalized harmonic sum. Table 6 demonstrates the main harmony assumptions in which parameter α is tuned according to domain attributes. Harmony assumption states that occurrences of event (term) t could be independent. It considers word burstiness and therefore could be an interesting candidate for QPP. Lets p_t be the probability of occurrence of term t, any arbitrary function $f()$ in $p_t^{f(n)}$ can be employed to represent a form of dependency over n occurrences of the term. In particular, $p_t^{a(n)}$ is the sequence probability where $a(n)$ is the assumption function. Equation (82) shows the new predictor as the sum of natural harmony weights for query terms.

$$\text{SumNaturalHarmony}(q) = \sum_{t \in q} 1 + \frac{1}{2} + .. \frac{1}{n(t,c)} \tag{82}$$

Natural harmony	$1 + \frac{1}{2} + ..\frac{1}{n}$	Harmonic sum
Alpha harmony	$1 + \frac{1}{2^{\alpha}} + ..\frac{1}{n^{\alpha}}$	Generalized harmonic sum
Square root harmony	$1 + \frac{1}{2^{(\frac{1}{2})}} + ..\frac{1}{n^{(\frac{1}{2})}}$	$\alpha = 1/2$; divergent
Square harmony	$1 + \frac{1}{2^2} + ..\frac{1}{n^2}$	$\alpha = 2$; convergent
Gaussian harmony	$\frac{2.n}{n+1}$	Explains the BM25-TF

Table 6: Main harmony assumptions [89].

DCBackgroundModel This predictor is the sum of DCM background model weights over the query terms. Let α_c be the background model, DCBackgroundModel is shown in equation (84).

$$\alpha_c(t) = m_c . \frac{df_t}{\sum_{i=1}^{n} |\overline{d_i}|} \tag{83}$$

$$\text{SumDCBackgroundModel} = \sum_{t \in q} \alpha_c(t), \tag{84}$$

$|\overline{d}|$ is the length of the distinct terms in document d, $n(t,d)$ is the within document term frequency and $|d|$ is the classic document length. The estimation of parameter mc is shown in equation (96).

4.3.3 Evaluation

Correlations between the predictors

I compared the correlations between pre-retrieval predictors including the novel candidates and the baselines with similar roots. These baselies are SumIDF (for the IDF-based baseline), SCS and MaxSCQ. The correlation task was performed by experimenting on 25 TREC medical and genomics topics using the *Pearson* coefficient. Not surprisingly, some features correlate with each other and others remain uncorrelated. The results reveal that the highest correlation is between SumAvgTF and SumNaturalHarmony. However, there are a few cases in the query set which do not follow this pattern. Although we have not calculated the statistical significance of coefficients, the correlation results may help us to decide which features are suitable candidates to potentially be combined for a performant prediction framework. Table 7 shows the correlations between the predictors which used in our experiments.
Another interesting observation is the strong degree of correlation between SumIDF, SumAvgTF and PosTF-IDF. This correlation shows the role of these features as exemplary discriminators. Moreover, our experimental results contradict the common assumption which relies on the effectiveness of the query length parameter in the prediction tasks. As an example, table 8 shows that although the query "gens are involved in insect segmentation?" has six terms, the SumIDF is evidently higher than the query "drugs are associated with lysosomal abnormalities in the nervous system" which consists

of ten words.

4.3.4 Discussion

It has long been recognized that a problematic obstacle to fully deliver IR models to applications of the same task is uncertainty in the prediction of their success rates [41, 58]. We have introduced a new family of pre-retrieval predictors based on word burstiness (inspired by DCM and Harmony assumption), position-based TF-IDF and Average Term Frequency. QPP plays an important role in the contribution of this thesis. SumIDF is used in the development of the aggregation parameter of FDCM (eq (104)(page 83) in section 4.4.2) and MaxIDF is used for the the corresponding RFDCM parameter (eq (147)(page 115) in section 6.2.2). Moreover, harmony assumption is considered in the feature set for training our IR recommender (eq (154)(page 117) in section 6.2.2).
This section aimed to provide formal grounds for developing a probabilistic framework which serves the query performance prediction purpose with respect to different IR models. We briefly discussed the influence of information needs on forecasting the retrieval quality concerning medical collections. Moreover, we employed a parameter derived from Dirichlet-multinomial Background Model and used the harmony assumption to develop new predictors which give rise to term dependency and burstiness. We compared the correlations between the proposed features and the well-established pre-retrieval predictors including SumIDF, SCS and MaxSCQ in order to explore some hidden features including burstiness, term dependency and term position which could impact the prediction quality. The highest correlation coefficient turned out to be between SumAvgTF and SumNaturalHarmony. Another interesting finding was the strong degree of correlation between SumIDF, SumAvgTF and PosTF-IDF. The results will help us to learn which predictors are worth being combined in order to increase the prediction accuracy.
Future work will look to evaluate the performance of the proposed predictors on the Medline citations. It could be interesting to detect the effective predictors and subsequently compute their efficiency in some other medical collections. In future work, we also aim to explore the role of Divergence from randomness (DFR) in QPP and discuss the relation between DFR, SCS and Natural Harmony.

Predictor	Predictor	Correlation Coefficient
SumAvgTF	SumNaturalHarmony	0.994
SumIDF	PosTF-IDF	0.860
SumIDF	SumAvgTF	0.811
SumIDF	SumNaturalHarmony	0.774
SumNaturalHarmony	SumDCBackgroundModel	0.691
PosTF-IDF	SumNaturalHarmony	0.683
SumAvgTF	SumDCBackgroundModel	0.646
SumIDF	SumDCBackgroundModel	0.404
PosTF-IDF	SumDCBackgroundModel	0.385
PosTF-IDF	MaxSCQ	0.319
SumAvgTF	MaxSCQ	0.237
PosTF-IDF	MaxSCQ	0.188
SumDCBackgroundModel	MaxSCQ	0.138
SumNaturalHarmony	MaxSCQ	00.090
SumIDF	SCS	-0.341
SumAvgTF	SCS	-0.450
PosTF-IDF	SCS	-0.468
SumNaturalHarmony	SCS	-0.500
SumDCBackgroundModel	SCS	-0.591

Table 7: Correlations between the pre-retrieval predictors.

Query	SumIDF	SumAvgTF	PosTF-IDF	SumNaturalHarmony	SumDCBackgroundModel	SCS	MaxSCQ
children with dental caries	6.4	6.4	12.1	37.8	0.3	6.5	29.0
Patients who developed disseminated intravascular coagulation in the hospital	12.0	10.5	18.4	92.3	1.5	-9.5	26.3
patients with inflammatory disorders receiving TNF-inhibitor treatments	8.0	8.1	13.7	68.1	2.7	-6.7	23.0
patients with acute tubular necrosis due to aminoglycoside antibiotics	12.4	11.3	15.0	92.0	3.5	-9.5	28.2
patients who presented to the emergency room with an actual or suspected miscarriage	15.7	11.5	24.1	103.2	3.5	-2.3	28.4
adult inpatients with Alzheimer disease admitted from nursing homes with pressure ulcers	17.0	14.5	21.3	116.3	5.8	-3.9	28.1
patients who have had a carotid endarterectomy	8.6	10.8	13.1	83.4	6.5	-13.8	29.1
patients with hearing loss	4.5	6.5	9.3	42.2	2.5	-8.1	27.1
hospitalized patients treated for methicillin-resistant Staphylococcus aureus (MRSA) endocarditis	19.1	13.0	20.3	77.0	3.0	-7.0	32.3
patients diagnosed with localized prostate cancer and treated with robotic surgery	15.2	13.6	25.5	103.7	3.8	-2.4	27.3
women with osteopenia	5.1	3.4	12.3	23.1	0.2	-1.3	28.6
adult patients who received colonoscopies during admission which revealed adenocarcinoma	16.2	14.9	27.2	131.2	8.1	-19.9	26.8
what serum proteins change expression in association with high disease activity in lupus?	14.3	15.8	16.0	139.4	5.4	-6.4	26.4
mutations in the Raf gene are associated with cancer?	6.0	7.4	9.1	59.3	1.5	4.8	24.7
drugs are associated with lysosomal abnormalities in the nervous system?	9.7	8.3	13.2	82.6	2.1	-1.9	26.5
cell or tissue types express receptor binding sites for vasoactive intestinal peptide (VIP) on their cell surface	17.2	18.6	20.5	152.2	4.2	-10.3	31.1
signs or symptoms of anxiety disorder are related to coronary artery disease	13.1	12.9	18.9	109.4	2.5	-5.3	26.4
toxicities are associated with zoledronic acid	5.1	4.3	8.2	41.8	1.6	-3.5	27.6
gens are involved in insect segmentation?	13.3	4.8	19.1	38.1	0.3	-0.3	28.7
in what diseases of brain development do centrosomal genes play a role?	17.1	12.5	27.2	114.8	2.3	-8.2	27.7
which anaerobic bacterial strains are resistant to Vancomycin?	8.1	8.0	12.9	69.1	3.1	-7.9	29.9
what viral gens affect membrane fusion during hiv infection?	17.0	14.2	29.5	97.9	2.8	-18.3	27.2
what pathways are involved in Ewing's sarcoma?	12.0	6.4	12.0	58.0	0.5	0.5	25.4
what tumor types are found in zebrafish?	9.9	7.42	12.7	63.4	2.4	-5.7	28.0
proteins make up the murine signal recognition particle	12.8	9.8	19.6	89.2	1.4	-13.3	26.2

Table 8: Numeric values for the pre-retrieval predictors regarding 25 medical and genomics TREC topics.

4.4 FDCM: Conceptual extension of DCM

4.4.1 Background

Despite the fact that concepts have been widely used in biomedical retrieval, studies often report limited or no improvement compared to well-known bag of words (BOW) methods [94]. AI has always been influential for semantic IR e.g. conceptual graphs [73]. More recently, complex approaches based on neural networks have received much attention. However, they might not be generalizable and need various requirements e.g. large memory. This section aims to build the grounds for establishing clear light-weighted concept-based standards to be used by data scientists. We introduce a balanced model based on Dirichlet Compound Model (DCM) as an effective extension of language modelling. Our model is developed upon the integration of concepts and terms. We compare the term-based model with its semantic generalized extension to investigate the level of required semantics for the combination. The complexity level of the model is between a recent advanced AI approach such as Bio-Bert [59] and a plain BOW model with no interpretation. This section evaluates the effectiveness of the model with term-based and concept-based language modelling baselines.
I consider DCM as an advanced extension of language modelling because it is probabilistically well-defined and shown to be performant [23].

Representations

BOW Representation is simply a transformation from sequential words in the query to a list. For the example query "popular drugs that successfully treat heart disease", the BOW representation is as follows:

```
BOW = [popular, drugs, that,
 successfully, treat, heart, disease]
```

To generate the so-called bag of concepts (BOC), we leverage the MetaMap ontology and the corresponding entity recognition tool. MetaMap is developed by the Lister Hill National Center for Biomedical Communications at the National Library of Medicine (NLM). It uses a knowledge-intensive technique, natural-language processing (NLP), and computational-linguistic techniques, and is used worldwide in industry and academia [2]. If we pass on

[2]https://www.nlm.nih.gov/research/umls/implementation_resources/metamap.html

a raw text to the MetaMap NLP tool, it will automatically turn on the NLP engine and extracts the related part-of-speech tokens. Subsequently, it gives us triggers, entity types and other required properties. The outcome of such a process for my example query is as follows (explanation of the fields in the outcome are included):

```
[{score= 8.34 ,
preferred_name="Heart Diseases",
cui="C0018799",
semtypes="[dsyn]",
trigger= "["HEART DISEASE,NOS"-tx-1-
            heart disease"-noun-0]",
location="TX",
tree_codes= "C14.280"}
,
{score= 5.18,
preferred_name="Therapeutic procedure",
cui="C0087111",
semtypes="[topp]",
trigger="["TREAT"-tx-1-"treat"-verb-0]",
location="TX",
tree_codes= "E02"}
,
{score= 3.68,
preferred_name= "Drugs - dental services",
cui="C3687832",
semtypes="[topp]",
trigger="["Drugs"-tx-1-"drugs"-noun-0]",
location="TX",
tree_codes="" ""}
,
{score= 3.68,
preferred_name= "Pharmaceutical Preparations",
cui= "C0013227",
semtypes= "[phsu]",
trigger= "["Drugs"-tx-1-"drugs"-noun-0]",
location="TX",
```

```
tree_codes= "D26")}
,
{score= 3.45,
preferred_name="popularity",
cui="C0679970",
semtypes="[socb]",
trigger="["popularity"-tx-1-"popular"-adj-0]",
location="X",
tree_codes=""}]
```

- **Score**: MetaMap Indexing (MMI) score has a maximum score of 1000.00, but this score varies based on the frequency of concepts and the length of a given text. The MMI ranking function was proposed by Aronson [5] which is based on the characterizing power or aboutness of medical concepts. Based on the algorithm, the higher the score is, the more relevant the UMLS concept is to the query. The MMI results are shown in highest to lowest order.

- **Preferred Name**: The preferred name for the UMLS concept is simply the generic title of the concept. For example, we have several terms for the concept 'Flu', but the so-called preferred name or the entity name is 'influenza virus.'

- **CUI**: Concept Unique Identifier.

- **Sem Types**: A comma-separated list of the semantic types associated with the concept, e.g. bpoc means Body Part, Organ, or Organ Component. The comprehensive list of MetaMap semantic types is provided in [3].

- **Trigger**: Trigger is a comma-separated list of the terms in the query which triggered the identification of medical concepts. This field is very useful in respect to IR since it helps for the determination of concept frequencies.

- **Loc**: Location of the trigger related to the concept within the query.

[3]http://metamap.nlm.nih.gov /SemanticTypesAndGroups.shtml

- **Treecode**: A semicolon-separated list of codes. Tree code is a branching structure of hierarchies with categories such as terms, organisms, diseases, drugs, chemicals, etc. A category has further subcategories, and the descriptors of each subcategory are listed hierarchically from most general to most specific within the structure.

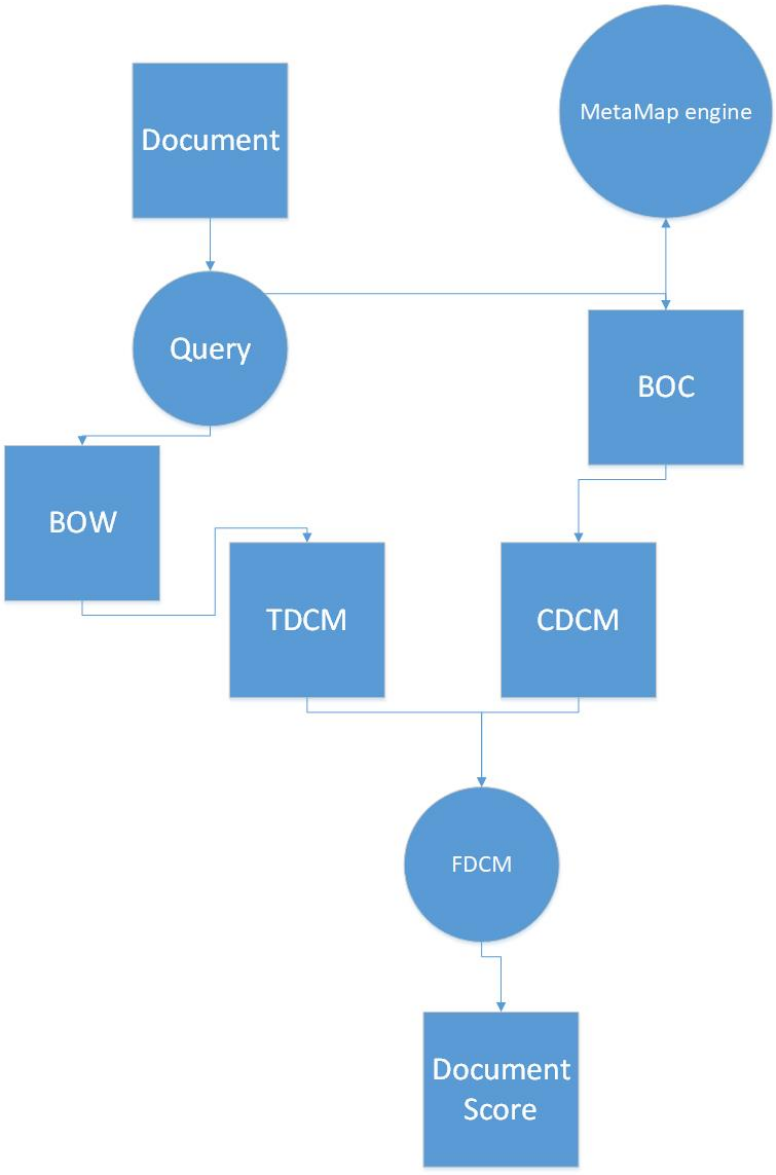
In addition to the entity type, which is derived from the "preferred name", we incorporate the concept frequency along with the MMI score into the BOC representation. The determination of the concept frequency is based on counting triggers of the concept in the given query. Below is the BOC representation of my example query:

```
[{'term': 'heart diseases', 'score': 8.34,
'type': 'dsyn', 'freq': '1'},
{'term': 'therapeutic procedure',
'score': 5.18, 'type': 'topp', 'freq': '1'}
, {'term': 'drugs - dental services',
'score': 3.68, 'type': 'topp', 'freq': '1'},
{'term': 'pharmaceutical preparations',
'score': 3.68, 'type': 'phsu', 'freq': '1'},
{'term': 'popularity', 'score': 3.45,
'type': 'socb', 'freq': '1'}]
```

In this example, there is no duplication in the query. Therefore, concept frequencies are set to 1. As can be seen, the concept "heart diseases" receives the highest MMI score, whereas the concept "Popularity" has the least score in the list. We also need to ensure all semantic types are biomedical and therefore I manually excluded non-biomedical types by introducing a stop-words-list which is shown below:

```
['qlco','qnco','fndg','ftcn',
'mnob','geoa','menp',
'inpr','tmco','cnce',
'prog','clna','aggp'
,'edac','idcn','spco',
'resa','acty','hlca',
'ocac','ocdi','ocac',
'socp'])
```

Figure 8: FDCM design.



As figure 8 shows, to rank a given document, we generate the BOW and BOC representations. I calculate TDCM and CDCM scores and pass them to the aggregator framework.

Term-based DCM (TDCM)

We define DCM as the probability distribution density function. Let θ be the W-dimensional probability simplex e.g. $\sum_t \theta_t = 1$ and let α be the Dirichlet entries (vector). The Dirichlet distribution based on the Bayesian hierarchical modeling framework is defined as follows:

$$p\left(\theta, \alpha\right) := \frac{\Gamma\left(\sum_t \alpha_t\right)}{\Pi_t \Gamma\left(\alpha_t\right)} \cdot \Pi_t {\theta_t}^{\alpha_t - 1} \tag{85}$$

When modeling texts, the vector θ represents the document [70]. By applying logarithm to the probability, we define the logarithmic Dirichlet distribution as follows:

$$\log p\left(\theta, \alpha\right) := \sum_t \left(\alpha_t - 1\right) \cdot \log \theta_t + \log \Gamma\left(\sum_t \alpha_t\right) - \left(\sum_t \log \Gamma\left(\alpha_t\right)\right) \tag{86}$$

The Dirichlet distribution is a distribution over probability vectors and there are multiple ways to represent the document as a probability vector. However, the rationale is to consider the θ vector as drawing a BOW from documents. Obviously the problem with this approach is that each document tends to contain only a small portion of the vocabulary, resulting in many of the entries being zero.
The other issue is the difficulty in smoothing because if even one entry in the Dirichlet likelihood is zero, it results in the zero probability.
A better approach is to hire multinomial distribution to model the documents where the parameters are based on the Dirichlet distribution. The likelihood probability of a document with the length of n is :

$$p\left(d, \alpha\right) := \int p\left(d, \theta\right) p\left(\theta, \alpha\right) \cdot d\theta \tag{87}$$

If we normalize the above equation, it could be rewritten as follows:

$$p\left(d,\alpha\right):=\frac{n!}{\Pi_t n\left(t,d\right)}\frac{\Gamma\sum\alpha_t}{\Gamma\left(\sum_t\alpha_t+n\right)}\cdot\Pi_t\left(\frac{\Gamma\left(n\left(t,d\right)+\alpha_t\right)}{\Gamma\alpha_t}\right) \tag{88}$$

The classical probability ranking principle suggests to rank documents by the log-odds ratio of their probabilities (the relevant class against the non-relevant class) [116]. Therefore the score of document d against query q is as follows:

$$\mathrm{RSV}\left(d,q\right):=\frac{p\left(d,q\right)}{p\left(d\right)} \tag{89}$$

By applying the arguments estimated in eq (88) into the above formula, we define the final ranking model as below:

$$\mathrm{RSV}\left(d,q\right):=\left(\sum_{i:n(t_i,d)>0}\sum_{j=0}^{n(t,d)-1}\log\frac{j+\omega_i}{j+\pi_i}\right)-\sum_{j=0}^{n-1}\log\frac{j+\sum_i\omega_i}{j+\sum_i\pi_i}, \tag{90}$$

Where ω is the Dirichlet vector for the relevant class and π is for the non relevant class. The above equation is the result of the division of relevant and non relevant probabilities:

$$\log\left(\frac{\frac{\Gamma\sum\omega_t}{\Gamma\left(\sum_t\omega_t+n\right)}\cdot\Pi_t\left(\frac{\Gamma(n(t,d)+\alpha_t)}{\Gamma\omega_t}\right)}{\frac{\Gamma\sum\pi_t}{\Gamma\left(\sum_t\pi_t+n\right)}\cdot\Pi_t\left(\frac{\Gamma(n(t,d)+\pi_t)}{\Gamma\pi_t}\right)}\right) \tag{91}$$

This score is derived from the probability of the document $p\left(d|q\right)$, However we can turn it around and estimate the probability of the query as shown in definition (4):

$$\frac{p\left(q|d\right)}{p\left(q\right)} \tag{92}$$

The estimation of the Dirichlet parameters for the likelihood formulas can be based on the moment theory or heuristic approaches could be hired. An iterative gradient descent optimization based on MLE method can be used to estimate the vector by computing the gradient of the DCM log-likelihood. The maximum likelihood estimate [70, 76] could also be computed using the fixed number of iterations.

$$\pi_i^{new} := \pi \cdot \frac{\sum_i \psi\left(n\left(t, d_i\right) + \pi_i\right) - \psi\left(n\left(t, d_i\right)\right)}{\sum_i \psi\left(n\left(d_i\right) + \sum \pi_i\right) - \psi\left(n\left(d_i\right)\right)} \tag{93}$$

The other approach for the determination of the non relevant vector is:

$$\pi_i := n\ P(t_i|c) := avgDL\ \frac{n(t_i, c)}{\sum_i n(t_i, c)} \tag{94}$$

Let $n_T(d, c)$ be the number of distinct terms in document d, the document model (relevant class) based on the Cummins estimation is:

$$\omega_i := n_T(d, c)\frac{n(t_i, d)}{|d|} = n_T(d, c) \cdot P(t_i|d) \tag{95}$$

Cummins etal. [23] brought a parameterization of the Dirichlet- multinomial distribution to language modelling. Let $|\,d\,|$ be the number of terms in document d, and let $|\,\overline{d}\,|$ be the number of distinct terms. The corresponding Dirichlet Compound document model is $\alpha_d(t) = (|\,\overline{d}\,| \cdot n(t, d))/|\,d\,|$, where $n(t, d)$ is the within document term frequency. The background model is $\alpha_c(t) = (m_c \cdot \mathrm{df}_t)/\sum_{i=1}^{n} \mathrm{df}_i$. The parameter m_c reflects term burstiness in collection c. The estimation of this parameter is as follows:

$$m_c = \frac{\sum_{i=1}^{N} |\,\overline{d_i}\,|}{\sum_{i=1}^{N} \psi\left(|\,d_i\,| + m_c\right) - n \cdot \psi(m_c)}. \tag{96}$$

Where N denotes the number of documents in the collection, and ψ is the digamma function $\psi(x) = \Gamma'(x)/\Gamma(x)$.
The experiments of *Cummins etal.* [23] suggest to initialize m_c (term burstiness) to the average document length. They showed that within 15 iterations the value of m_c will converge. Let q be a query, d a document and c a collection. Then, the TDCM weight and RSV are defined as follows:

$$\mathrm{W}_{\mathrm{TDCM}}\left(t,d,q,c\right) := \mathrm{TF}\left(t,q\right)\ \log\left(\left(1-\lambda_d\right)+\lambda_d\cdot\frac{\alpha_d(t)}{\alpha_c(t)}\right) \tag{97}$$

Definition .6 (Term-only DCM (TDCM))

$$\mathrm{RSV}_{\mathrm{TDCM}}(d,q,c) := \sum_{t\in d}\mathrm{W}_{\mathrm{TDCM}}(t,d,q,c) \tag{98}$$

The mixture (smoothing) parameter λ_d can either be a constant and independent from the document (Jelinek-Mercer mixture) or be proportional to the document model. In respect to the multinomial language model, an effective smoothing method is a Dirichlet prior $\lambda_d := \mu/(\mu + |\,d\,|)$. It has been shown that it results in a relatively consistent and stable performance [119].

Concept-based DCM (CDCM)

Concepts are entities in the corpus which are instantiated from super categories (e.g. symptoms, signs and drugs for the medical domain). We introduce an independent concept-based model derived from DCM and explain how to integrate it with the term-based quantification explained in the previous section.
Processing all concepts equally is not rational and would result in weak retrieval. We address this issue by reinforcing the conceptual model with a probability factor based on the MMI scores of MetaMap. The MMI ranking function was designed by Aronson [5] to indicate the characterizing power or aboutness of a given concept for a piece of text. Our experiments show that the use of the ratio of the concept-score to the total score of concepts in the query improves the performance:

$$\mathrm{CF}_{\mathrm{score}}\left(\varphi,q\right) := \frac{\mathrm{score}_{\mathrm{mmi}}\left(\varphi,q\right)}{\sum_{\varphi'\in q}\mathrm{score}_{\mathrm{mmi}}\left(\varphi',q\right)} \tag{99}$$

where φ is a concept and $\mathrm{score}_{\mathrm{mmi}}(\varphi,q)$ is the Metamap score of the concept against the query. The use of a saturating function [88] would be the alternative approach for the concept frequency, but since queries are normally very short compared to the documents, its application is controversial.

For CDCM, the steps discussed in TDCM were followed with the difference being that parameters (e.g. m_c) are based on concepts instead of terms.

$$\mathrm{W}_{\mathrm{CDCM}}\left(\varphi, d, q, c\right) := \mathrm{CF}_{\mathrm{score}}(\varphi, q) \cdot \log\left((1-\omega_d) + \omega_d \cdot \frac{\alpha_d(\varphi)}{\alpha_c(\varphi)}\right) \tag{100}$$

where ω_d is the mixture parameter. The CDCM weight in eq (100) is for concepts what the TDCM weight in eq (97) is for terms. Similar to the determination of parameter λ_d for the TDCM part, the parameter w_d for CDCM is proportional to the document length and could be calculated by regulating parameter μ. However, for ω_d, the length of the document is based on the number of observed concepts in the document. Moreover, $\alpha_d(\varphi)$ and $\alpha_c(\varphi)$ are the respective Dirichlet parameters for the concept φ. Following eq (98) for the TDCM score, the CDCM score is:

Definition .7 (Concept-only-only DCM (CDCM))

$$\mathrm{RSV}_{\mathrm{CDCM}}(d, q, c) := \sum_{\varphi \in d} \mathrm{W}_{\mathrm{CDCM}}(\varphi, d, q, c) \tag{101}$$

4.4.2 Contribution

Full DCM (FDCM)

In this section we define the final model constructed from the term-based and conceptual approaches. This study is limited to the use of terms and concepts, however any other semantic predicates such as sentiments or relationships could be consolidated into the model. We name the proposed model Full Dirichlet Compound Model (FDCM), where σ_T is the respective aggregation (mixture) parameter:

Definition .8 (Full DCM (FDCM))

$$\begin{aligned}&\mathrm{RSV}_{\mathrm{FDCM}}\left(d, q, c\right) := \\ &\quad \sigma_T \cdot \mathrm{RSV}_{\mathrm{TDCM}}(d, q, c) + (1-\sigma_T) \cdot \mathrm{RSV}_{\mathrm{CDCM}}(d, q, c)\end{aligned} \tag{102}$$

Given example query *"HIV and the GI tract, recent reviews"*, bag of words (BOW) is [HIV, GI, tract, recent, reviews] and bag of concepts (BOC) is

[HIV, GI tract]. As can be seen concepts have been occurred in both models. Furthermore, GI tract is recognized as a compound concept (better semantic representation) contrary to the BOW approach. Therefore, the use of CDCM results in boosting the impact of concepts on the overall FDCM score. Compared to traditional IR, where each term is considered only once, FDCM considers concepts *twice*.
Basically, the concepts have a higher impact on the score, since they are considered twice, in the TDCM score, and in the CDCM score.
Main contributions to follow will be RFDCM (Recommendation FDCM, section 6.2.1) and URFDCM (user-based RFDCM, section 6.2.2). While FDCM combines term-based (TDCM) and concept-based (CDCM) scores, RFDCM combines item-based (IDCM) and concept-based scores, and URFDCM combines user-based and concept-based scores.

Term-Concept Aggregation Parameter σ_T

The idea was to investigate the use of query performance predictors (QPPs) for estimating the FDCM mixture parameter σ_T. The well-established QPPs are discussed in [7]. I investigated the effectiveness of some semantic QPPs such as sum of IDFs, Simple Clarity Score (SCS), and average TF.

Parameter Setting/Shape

In this section, a semantic QPP is considered as the difference of term-based and concept-based values for the QPP.
As figure 11 shows, the difference of the sum of term-based IDFs and sum of concept-based IDFs is the most correlated predictor in in the CDCM improvement. Interestingly, our results show that sum of concept-based IDFs are significantly bigger than sum of term-based IDFs for top ranked items in OHSUMED. The IDF definition originates from the BM25 retrieval model. Let N be the number of documents, and $\mathrm{df}(x)$ be the number of documents in which concept (or term) x occurs.

$$\mathrm{IDF}(x) := \log\left(\frac{N - \mathrm{df}(x) + 0.5}{\mathrm{df}(x) + 0.5}\right) \tag{103}$$

We define the semantic information $\mathrm{q}_{\text{sem-info}}$ associated with the query as the difference between the sum of concept-based IDFs and the sum of term-based

IDFs:

$$\mathrm{q}_{\text{sem-info}} := \left(\sum_{\varphi \in \text{q-concepts}} \mathrm{IDF}(\varphi) \right) - \left(\sum_{t \in \text{q-terms}} \mathrm{IDF}(t) \right) \tag{104}$$

The value interval is $-\infty < \mathrm{q}_{\text{sem-info}} < +\infty$, where due to the logarithmic nature, values are on the smaller side (between -100 and 100). A positive value indicates that the discriminativeness of the semantic concepts in the query is higher than the discriminativeness of the terms (non-concept words, i.e. q-concepts and q-terms are disjoint). The histogram in figure 9 shows the number of OHSUMED queries that lie within the range of $\mathrm{q}_{\text{sem-info}}$ values that define the bin centre. Also, the cumulative frequency distribution (percentage) of queries against $\mathrm{q}_{\text{sem-info}}$ values represented in figure 10. Each term in the query can be linked to several MetaMap concepts which results in the high values for the sum of conceptual IDFs.

Figure 9: Histogram displaying number of queries over $\mathrm{q}_{\text{sem-info}}$ on OHSUMED.

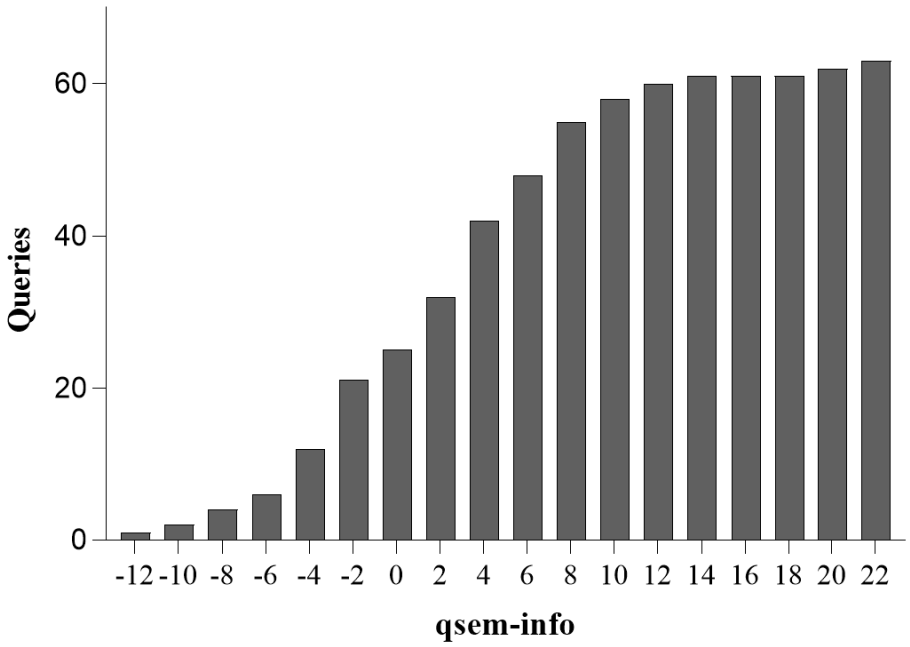
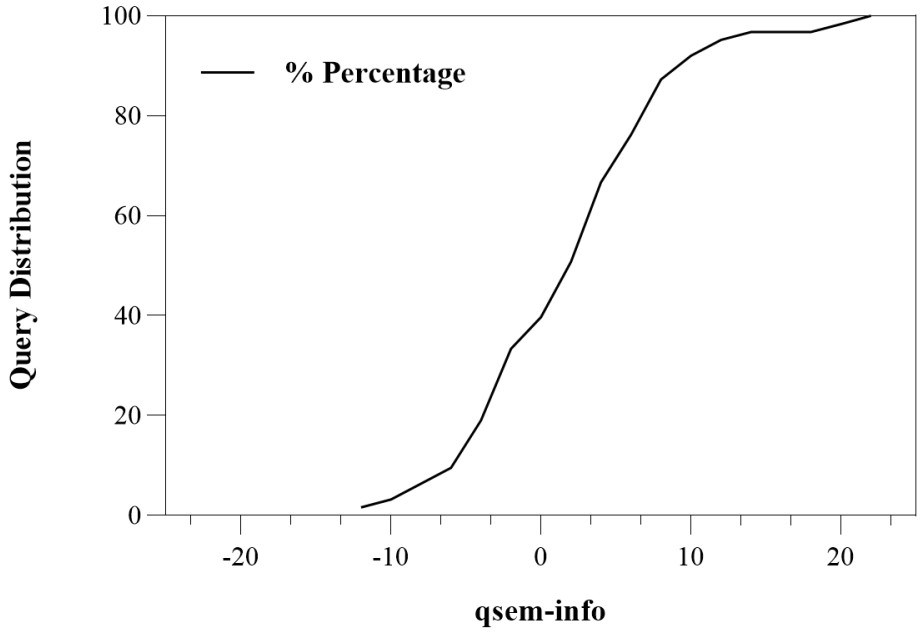


Figure 10: Cumulative frequency distribution of OHSUMED queries by $\mathrm{q}_{\text{sem-info}}$ values.



The semantic information (also referred to as the semantic strength) is input to estimating the aggregation parameter σ_T for aggregating the TDCM and CDCM scores in eq (102). The constraint is $0 \leq \sigma_T \leq 1$. The most common mathematical approach to satisfy the constraint is the sigmoid function $e^x/(1+e^x)$, where $x = \mathrm{q}_{\text{sem-info}}$. Alternatively, we apply $x/(1+|\,x\,|)$ which is inspired by a major IR concept, namely the BM25-TF quantification, but different to the case of the BM25-TF, the semantic query frequency can be negative. The semantic query frequency qsf is:

$$\mathrm{qsf} := \frac{\mathrm{q}_{\text{sem-info}}}{1+|\,\mathrm{q}_{\text{sem-info}}\,|} \tag{105}$$

This saturating functions maps the semantic information $\mathrm{q}_{\text{sem-info}}$ to the interval $(-1;+1)$. Such interval is commonly used for sentiment-based retrieval. The mapping is inspired by the saturating BM25-TF quantification $\mathrm{tf}(t,d)/(\mathrm{tf}(t,d)+\mathrm{pivdl})$, where for a document of average length, the pivoted

document length is pivdl = 1. For IR it is well understood that the saturating nature is very important for avoiding the dominating effect of relatively large numbers (frequencies).
Next, we define a normalised semantic query frequency, i.e. we transform the saturated frequency qsf to the internal $[0; 1]$. The min-max normalisation yields qsf_{norm}:

$$\text{qsf}_{\text{norm}} := \frac{\text{qsf} - \min}{\max - \min} \tag{106}$$

where $0 \leq \text{qsf}_{\text{norm}} \leq 1$. Note that due to the min-max normalisation values stretch over the full value interval.
Common in IR are two notions of query length: $|q|$ is simply the number of words (the raw query length), and $\text{q\_idf\_len} := \sum_{t \in q} \text{IDF}(t)$ is the IDF-based query length (a query predictor that indicates how easy it will be to find relevant documents).
Consequently, there are two approaches to obtain a *semantic query length*:

$$\text{q}_{\text{raw-sem-length}} := \text{qsf}_{\text{norm}} \cdot \text{q\_raw\_len} \quad (= \text{qsf}_{\text{norm}} \cdot |q|) \tag{107}$$

and

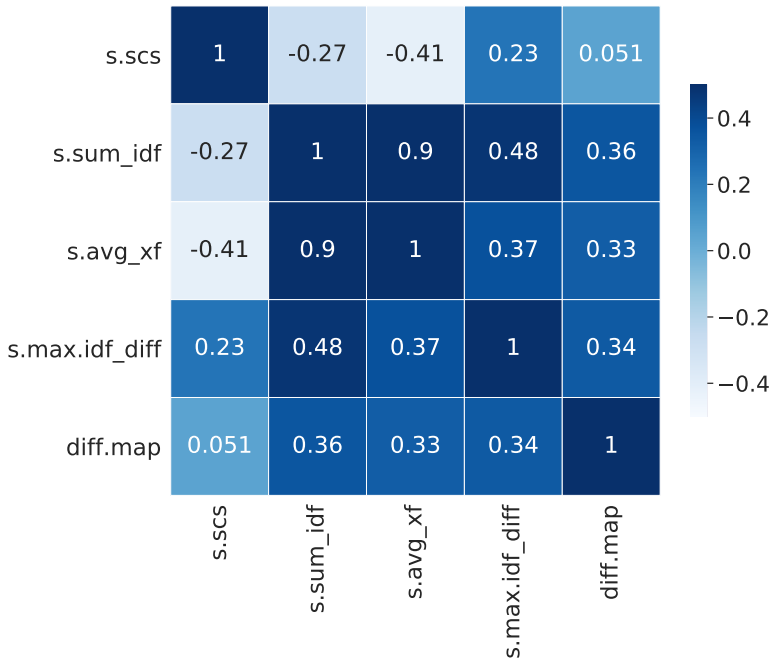
$$\text{q}_{\text{idf-sem-length}} := \text{qsf}_{\text{norm}} \cdot \text{q\_idf\_len} \quad \left(= \text{qsf}_{\text{norm}} \cdot \sum_{t \in q} \text{IDF}(t) \right) \tag{108}$$

The value interval is $0 \leq \text{q}_{\text{idf-sem-length}} < \infty$.
We have defined a family of parameters that feeds into candidates for σ_T.

Candidates of Aggregation Parameter σ_T

Parameters such as the semantic information $\text{q}_{\text{sem-info}}$, the semantic frequency qsf (the BM25-inspired saturation of $\text{q}_{\text{sem-info}}$), the raw semantic length $\text{q}_{\text{raw-sem-length}}$, and the idf semantic length $\text{q}_{\text{idf-sem-length}}$ feed into candidates for the FDCM mixture parameter σ_T.
Basically, we are investigating options of a generalised logarithmic discounting function $n/((n-1) + \log_2(1+x))$, where n is a shape parameter. The log-discount is small for high value of x, i.e. small if the semantic information is high.
The idea was to find out to what extent the use of semantic information parameters is beneficial. We focused on two instances inspired by the function in Paik's model [78].

Figure 11: Semantic QPP correlation matrix: Correlation between differences of concept-based and term-based query performance prediction values (semantic QPP variants) with the effectiveness of CDCM over TDCM; MAP $(CDCM)$ − MAP $(TDCM)$.



Definition .9 (Candidates for aggregation parameter σ_T)
Let n be an integer variable; if allows for optimising the shape of the respective function.

$$\sigma_{T,\text{qf-sigmoid},n}(q) := \frac{e^{(n+\text{qf})}}{1+e^{(n+\text{qf})}} \tag{109}$$

$$\sigma_{T,\text{raw-sem},n}(q) := \frac{n}{(n-1)+\log_2\left(1+\text{q}_{\text{raw-sem-length}}\right)} \tag{110}$$

$$\sigma_{T,\text{idf-sem},n}(q) := \frac{n}{(n-1)+\log_2\left(1+\text{q}_{\text{idf-sem-length}}\right)} \tag{111}$$

Equation (109) is sigmoid-based and considers the semantic query frequency, which is based on sums of IDFs, but is relatively independent from the raw query length $|q|$ (word count in q).
Differently, eq (110) leverages a logarithmic component in which the query factor qsf_{norm} is multiplied (boosted) with the raw query length $|q|$. Our experiments suggest that the raw query length is not big enough to perform well within the logarithm. This observation led to the candidate in equation (111), where the semantic query length from (108) is applied and therefore, $\sigma_T := \sigma_{T,\text{idf-sem},n=1}$. The evaluation of the candidates is discussed in section 4.4.3.

4.4.3 Evaluation

Experimental setup

- **Implementation:** To generate the knowledge-oriented representation, we passed raw abstract texts of Medline documents to MetaMap and indexed lists of concepts accompanied by their frequencies, semantic types and scores. We counted 'trigger' attributes of MetaMap outputs to capture the corresponding frequencies. To perform the query formulation, we used MetaMap for the concept extraction and transformed the query strings into the same knowledge-oriented forms used for documents. The formulated query consists of a BOW representation and a hierarchy-structured representation including relevant concepts with their frequencies, semantic types and scores. We have not used any stop words in our experiments at all furthermore, Porter stemming was applied to all collections and queries.

Using the Lucene framework and the TLM with Dirichlet Prior, we indexed the data and retrieved pseudo relevant documents. Subsequently, top 100 documents were re-ranked by the models used in the experiments. To determine the Dirichlet parameters of conceptual and term-based DCMs we used $\mu = 2000$. Additionally, 15 iterations were used to estimate parameter m_c.

- **Data:** We selected three data collections for evaluation, TREC Genomics 2004, TREC Genomics 2005 and OHSUMED. The OHSUMED is a collection constructed from 348,566 references from MEDLINE based on 270 medical journals. For experiments on OHSUMED, we used 65 topics hired in TREC-9 Filtering Track. The test collection used in both TREC Genomics 2004 and TREC Genomics 2005 is a 10 years subset of the Medline provided a total of 4,591,008 citations. Each of the Genomics tracks consists of 50 distinguished topics. In all experiments only the 'full statement information need' portions from the topics were used.

- **Baselines:** I evaluate FDCM against LM baselines (since FDCM is LM-oriented) however, for the sanity check, I use the traditional TF-IDF baseline too. Since FDCM is based on a preferment instance of Poisson (as discussed in chapter 3), I do not evaluate the FDCM performance against other Poisson models. Term-based baselines include TLM with Dirichlet Prior smoothing [84], TDCM [23] and TF-IDF. Moreover, Concept-based baselines including CRM [109], concept-based DCM (CDCM) and language modelling CLM [74].

  CRM is a state of the art conceptual LM which gives rise to the importance of semantic types associated with concepts. It leverages an elite set of important semantic types constructed from pseudo relevant documents. [109] used indri to index data and combined their proposed model with inference network approaches. However, we used only the retrieval model introduced in their section and implemented it based on our own experimental settings. We developed the elite set based on the subset of semantic types used in the top 10 pseudo documents.

Aggregation Parameter σ_T: Evaluation and Analysis

We performed an extensive study to compare the performances of the FDCM candidates generated from variations of parameter σ_T. The parameter depends on the query q and the collection c (for the IDFs of the terms). For concise formulation, we apply $\sigma_T := \sigma_T(q)$ where the parameters q and c can be omitted where the context is clear.
All candidates have decent quality, however, our study showed that σT, idf-sem is more stable across different benchmarks as shown in table 9. The table compares the performances of the FDCM candidates generated from variations of parameter σ_T. Also, the study revealed that damping the effect of the qsf for $n > 1$ in $\sigma_{T,\text{idf-sem},n}(q)$ is not effective. Therefore we set the aggregation parameter σ_T as follows:

$$\sigma_T \quad := \quad \sigma_{T,\text{idf-sem},n=1}(q) \qquad \left(= \frac{1}{\log_2\left(1 + \mathrm{q}_{\text{idf-sem-length}}\right)} \right) \qquad (112)$$

The greater the semantic query length $\mathrm{q}_{\text{idf-sem-length}}$, the less is σ_T, i.e. the less is the impact of the TDCM score, and the higher is the impact of the CDCM score on the FDCM score. Neither the normalized query-length parameter $\sigma_{T,\text{qf-sigmoid},n}$ nor $\sigma_{T,\text{raw-sem},n}$ which gives significant rise to the qf are shown to be consistently effective.

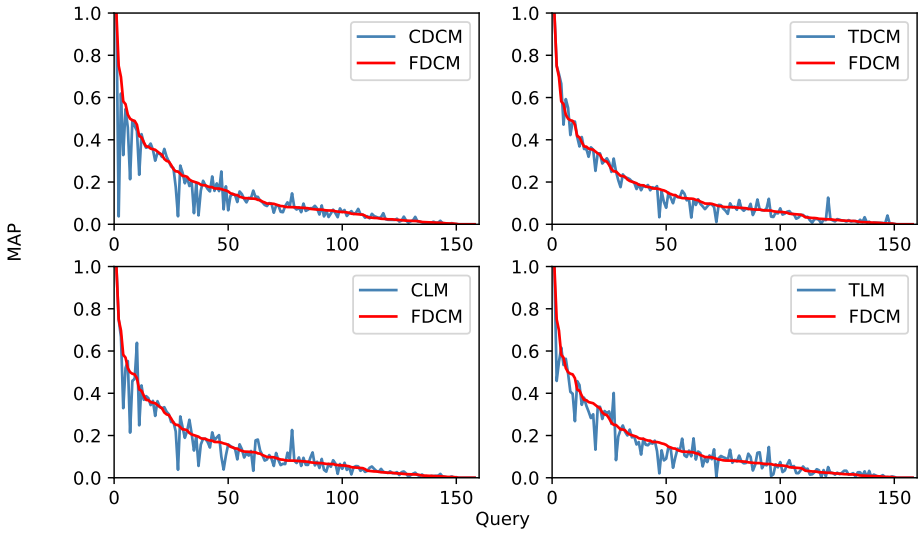
FDCM: Evaluation

Table 10 shows the experimental results used for comparing FDCM with baselines using three metrics including Reciprocal Rank (R-Rank), Mean Average Precision (MAP) and nDCG. It is common to choose a few IR metrics for evaluation, and I used these metrics since they are more precision-oriented. However, future work could investigate further the evaluation of FDCM for specific Recall-based tasks. The main finding could be that FDCM consistently outperformed well-known term-based baselines derived from language modelling and performed better than both TDCM and CDCM in all datasets. As expected, the combined model was shown to be more effective than the concepts-only baselines. We also conducted the paired t-test with $p < 0.05$ to compute the significance of improvements. The plots in figure 12 showed the query-based MAP comparison of FDCM with TLM, CLM, TDCM and CDCM. Some queries significantly worked better with FDCM. Our investigation suggest all these queries are fact-based and longer than average queries.

σ_T	**OHSUMED**		**TREC 2004**		**TREC 2005**	
	MAP	**nDCG**	**MAP**	**nDCG**	**MAP**	**nDCG**
$\sigma_{T,\text{raw-sem},1}$	0.214	0.289	0.128	0.191	0.200	0.231
$\sigma_{T,\text{raw-sem},2}$	0.212	0.289	0.133	0.196	0.200	0.228
$\sigma_{T,\text{raw-sem},3}$	0.213	0.290	0.143	0.202	0.200	0.228
$\sigma_{T,\text{idf-sem},1}$	**0.223**	**0.294**	**0.155**	**0.217**	**0.203**	**0.231**
$\sigma_{T,\text{idf-sem},2}$	0.221	0.293	0.143	0.200	0.201	0.228
$\sigma_{T,\text{idf-sem},3}$	0.219	0.291	0.143	0.199	0.201	0.228
$\sigma_{T,\text{qf-sigmoid},0}$	0.210	0.289	0.134	0.196	0.196	0.224
$\sigma_{T,\text{qf-sigmoid},1}$	0.213	0.290	0.145	0.201	0.193	0.222
$\sigma_{T,\text{qf-sigmoid},2}$	0.214	0.290	0.142	0.198	0.192	0.224

Table 9: Performances of the models derived from the regulation of parameter σ_T: $\sigma_{T,\text{idf-sem},1}$ is shown to be the most effective combination parameter.

Figure 12: MAP analysis of the models sorted by FDCM performance: The main finding is that some queries worked significantly better with FDCM. For a few queries, FDCM could not outperform the baseline. However, in most of these cases the MAP difference was not high.



Another finding is that when FDCM could not outperform the baseline, in most of the cases the MAP difference was not high.

Model	OHSUMED			TREC 2004			TREC 2005		
	R-Rank	MAP	nDCG	R-Rank	MAP	nDCG	R-Rank	MAP	nDCG
TLM	0.686	0.208	0.289	0.650	0.135	0.202	0.512	0.176	0.218
TDCM	0.683	0.211	0.286	0.693	0.153	0.214	0.467	0.191	0.220
TF-IDF	0.545	0.173	0.259	0.618	0.123	0.189	0.490	0.179	0.214
CLM	0.671	0.201	0.284	0.704	0.147	0.214	0.507	0.199	0.225
CDCM	0.670	0.203	0.283	0.655	0.130	0.201	0.533	0.183	0.225
CF.IDF	0.587	0.182	0.267	0.578	0.134	0.205	0.520	0.178	0.220
CRM	0.491	0.165	0.250	0.619	0.145	0.212	0.410	0.164	0.204
FDCM	$\mathbf{0.724}^{\gamma\delta}$	$\mathbf{0.223}^{\theta\alpha\gamma\delta}$	$\mathbf{0.294}^{\zeta\alpha\gamma\delta}$	$\mathbf{0.750}^{\beta\alpha\gamma\delta}$	$\mathbf{0.155}^{\beta}$	$\mathbf{0.217}^{\beta}$	$\mathbf{0.555}^{\zeta\delta}$	$\mathbf{0.200}^{\delta}$	$\mathbf{0.231}^{\beta\zeta\delta}$

Table 10: Ranking performances of the FDCM (with $\sigma_{T,\text{idf-sem},1}$) and the baseline methods: The bold font denotes the best result in that evaluation metric. β, θ, ζ , α, γ and δ indicate statistically significant improvements over LM$^{\beta}$, CLM$^{\theta}$, TDCM$^{\zeta}$, CDCM$^{\alpha}$, CF-IDF$^{\gamma}$ and CRM$^{\delta}$. The statistically significance is based on the paired t-test with p-value <0.05.

4.4.4 Discussion

Semantic (concept-based) retrieval approaches are important in biomedical IR and other domains. Purely conceptual DCM (CDCM) captures semantics, but CDCM needs to be combined with term-based DCM (TDCM). This section presented a semantic extension of TDCM for dealing with the problem of varying performance of concept-only IR. The transformation from TDCM over CDCM to FDCM utilises a framework that firstly separates terms and concepts, and then combines them into an overall FDCM score. One of the main contributions of this section was the IDF-motivated estimation of the mixture parameter for the proposed FDCM. We compared a set of aggregation candidates. The experimental results show that the best candidate was $\sigma_{T,\text{idf-sem},1}$ which is based on a logarithmic adjustment of the impact of TDCM and CDCM. Moreover, FDCM consistently outperformed

the language modelling baselines and was shown to be more effective than both terms-only and concepts-only DCM models. In conclusion, this work provides a bridge between IR and Artificial Intelligence in the sense that AI is knowledge-oriented and this work transparently integrates a basic NLP technology (also knowledge-oriented) with IR. Moreover, this works facilitates the development of clear recipes to understand and establish standards. It is important because many applications such as medical and criminal domains use standards to take advantage of the retrieval technology, especially when the queries are semantic.

Future work

FDCM is based on aggregating retrieval scores of the semantic dimensions. Combining individual weights of parings of terms and concepts is an interesting research direction for future work. Let σ_t be a term-based aggregation parameter, equation below represents a FDCM instance based on semantic pairings:

$$\begin{aligned} &\mathrm{RSV}_{\mathrm{FDCM}_{pairing}}(d,q,c) := \\ &\sum_{t\in q,\varphi\in q} \sigma_t \cdot \mathrm{W}_{\mathrm{TDCM}}(t,d,q,c) + (1-\sigma_t)\cdot \mathrm{W}_{\mathrm{CDCM}}(\varphi,d,q,c) \end{aligned} \tag{113}$$

Future work also investigates the effectiveness of new candidates for the aggregation parameter σ_T. Explanations of some possible candidates are detailed below:

- Let qf be the normalized query factor derived from the difference of the sum of IDFs of concepts and terms, the below candidate is based on document length:

$$\mathrm{ql}_n := \frac{n}{(n-1)+\log_2(1+|d|\cdot \mathrm{qf})} \tag{114}$$

- The use of MAX SCQ to boost the query factor. It is rational to increase the effect of CDCM when the conceptual MAX SCQ is high.

$$\mathrm{ql}_n := \frac{n}{(n-1)+\log_2(1+\mathrm{SCQ}\cdot \mathrm{qf})} \tag{115}$$

- The use of probability functions based on information theory e.g. entropy function:

$$\sigma_{\cdot} = -\mathrm{qf} \cdot \log_b \mathrm{qf} \tag{116}$$

- Leverage the ontology (Meta-Map) for aggregation. The mmi score of the concepts can be used:

$$\sigma_T := \left(\sum_{\varphi} \frac{Score_{mmi}(\varphi)}{\sum_{\acute{\varphi} \in q} Score_{mmi}(\acute{\varphi})} \cdot \mathrm{IDF}(\varphi) \right) - \sum_{t} \mathrm{IDF}(t) \tag{117}$$

  This candidate is based on the ratio of the concept score to the sum of scores in a given query and multiplying the result by the normal IDF.

Chapter 5

Feelings of Sentiment in IR

5.1 Chapter overview

Although sentiment analysis has received much attention in IR and other domains including NLP and text mining [67], incorporating the sentiment of words into IR models is still challenging and as yet no widely accepted standard exists for this task. The contribution of this chapter is a generalizable framework for quantifying term frequency variants with sentiments. The consideration of opinions is critical when it comes to the identification of the urgency level in notification filtering. IR models take collection-related features such as term rareness into consideration to rank documents, however they do not capture opinions in the retrieval process. For example concerning movie reviews, the word *'good'* might occur nearly in every review and from an IDF-point of view it is not informative and selective. We expect a query such as *'good comedy movie'* to find good movies, but IR might process it similar to *'comedy movie'* due to the high frequency of the term *good* in the collection.

This chapter attempts to address the above issue by employing two different tasks. Firstly, we add IDF of sentiment-bearing words as a notion of rareness to the sentiment classification process. Secondly, we generalize IR models by proposing intensity-aware methods which take sentiment intensity into consideration. The idea is to regulate Term Frequency quantification by boosting weights of sentiment-bearing words. Such boosting is expected to overcome the problem caused by the rareness of these words with respect to IR models.

We propose models derived from the strength of lexical features to improve sentiment-based ranking and to build the formal grounds for the development of a framework which incorporates opinions into relevance-based ranking. The results of the tests on movie reviews show that the use of IR parameters improves the quality of the sentiment classification. We show that opinion-aware LM and TF-IDF$_{\text{BM25}}$ outperform their corresponding basic models and perform better than the other candidates concerning polarity-based retrieval. Furthermore, we perform a query-based evaluation across a range of negative and positive queries to explore which features would improve the performance of the models. This chapter shows a pathway to achieve new standards of TF-IDF and LM for sentiment and intensity-aware retrieval.

5.2 Opinion-aware models

5.2.1 Background

Preliminaries

VADER Sentiment A sentiment lexicon is a dictionary of lexical features that are classified into positive and negative categories based on their semantic structures [68]. Polarity-based Lexicons categorize lexical features into binary classes, whereas intensity lexicons assign valence scores to the features. ANEW, SentiWordNet and SenticNet are commonly used intensity-based lexicons [53]. On the other hand, LIWC and Harvard GI are popular polarity-based lexicons that are used widely in social media domains [29, 54, 101, 82].
The well-known polarity-base lexicons suffer from two major deficiencies which are addressed by VADER rule-based algorithm [53]. Firstly, they are not able to deliver sentiments to lexical items common in social media such as acronyms, initialisms, emoticons or slang which are important components for sentiment analysis [24]. Secondly, they do not consider the intensity of sentiments. For example, "The food here is amazing" delivers more positive intensity than "The food here is good". LIWC would assign the same score to both sentences. Based on the VADER lexicon, a lexical feature could have a score between +1 and -1. Eq (118) shows how VADER calculates the compound sentiment of a sentence:

$$\text{sentiment}_{\text{vader}}(s) := \frac{\sum_{t \in s} \left(\sum_{i=1}^{n_L(t,s)} \text{W}_{\text{sentiment}}(t,i,s) \right)}{\sqrt{\left(\sum_{t \in s} \left(\sum_{i=1}^{n_L(t,s)} \text{W}_{\text{sentiment}}(t,i,s) \right) \right)^2 + \alpha}}, \quad (118)$$

where $\text{W}_{\text{sentiment}}(t,i,s)$ is the sentiment score of the i_{th} occurrence of term t, s is a sentence and α is a normalization parameter.
The algorithm is reinforced by five heuristics including punctuation marks, capitalization, modifiers, negation and 'but' checker. It regulates the score of a word depending on the distance between the word and its degree modifier. For example, a booster like *'Amazingly'* in the sentence *'I am Amazingly happy'* adds 0.293 to the sentiment score of *'happy'*, whereas if we replace the booster with a dampener like *'kinda'* the sentiment would be subtracted by 0.293. Moreover, the model analyzes the tri-gram before the base word to capture the negation by hiring a set of negator words like *'not'*. If a term is a degree modifier (either a dampener or booster), we call it an influencer. The equation below shows how the primary sentiment of a term is transformed into a sentence-dependant sentiment.

$$\text{W}_{\text{sentiment}}(t,i,s) := \acute{\text{W}}_{\text{sentiment}}(t) + seed(i,s) \quad (119)$$

$$\text{seed}(i,s) := \sum_{j=0}^{i-1} \alpha(j,i,s) \quad (120)$$

$$\alpha(j,i,s) = \begin{cases} 0 & \text{if } t_j \text{ is not an influencer} \\ \text{W}_{\text{f}}(t(j), i-j) & \text{if } t_j \text{ is an influencer} \end{cases}, \quad (121)$$

$\acute{\text{W}}_{\text{sentiment}}$ and *seed* are VADER internal functions where $\acute{\text{W}}_{\text{sentiment}}(t)$ is the primary sentence-independent sentiment score of term t, $seed(t,s)$ is a weight to be added to $\acute{\text{W}}_{\text{sentiment}}$ to bring the influencers (boosters and dampeners) into consideration. W_{f} estimates the weight of a single heuristic parameter in relation to the term by considering the distance and the constant weight of the parameter which is defined by VADER.
As eq(122) shows, the intensity of a lexical feature is the sum of the absolute value of the sentiment and it's corresponding seed weight.

$$\text{W}_{\text{intense}}(t,i,s) := \left| \acute{\text{W}}_{\text{sentiment}}(t) \right| + seed(i,s). \quad (122)$$

We consider the VADER rule-based algorithm as the baseline model to evaluate our sentiment-based methods. In order to rank the documents using VADER we hire the equation below.

$$\mathrm{RSV}_{\mathrm{vader}}(d) := \sum_{s \in d} \mathrm{sentiment}_{\mathrm{vader}}(s). \tag{123}$$

The RSV is high if the document contains many positive opinion words.

5.2.2 Contribution

Opinion-Aware TF

The main contribution of this section is the integration of opinions including sentiments and intensities with IR models. The novel $\mathrm{TF}_{\mathrm{total}}$ variants are shown in Definition 10. Moreover, we discuss the opinion-aware pivoted term frequencies and their corresponding $\mathrm{TF}_{\mathrm{BM25}}$ weights.

Opinion-aware $\mathrm{TF}_{\mathrm{total}}$: We introduce micro and macro models for the sentiment-based retrieval. Eq (124) shows the sentiment-based macro $\mathrm{TF}_{\mathrm{total}}(t, d)$, where $n_L(t, s)$ is the number of locations in which term t appeared in sentence s and $\mathrm{W}_{\mathrm{sentiment}}(t, i, s)$ returns the VADER score of i_{th} occurrence of term t in the sentence. In micro $\mathrm{TF}_{\mathrm{total}}$, we determine the term frequency independently and then multiply the result by the corresponding primary sentiment. Therefore, this model does not consider the impacts of degree modifiers. Eq (125) shows the micro $\mathrm{TF}_{\mathrm{total}}$.
The opinion-aware TF can also be adopted from intensity or force of lexical features. In this thesis, we considered the combination of strength and the corresponding seeds to determine the intensity weight $\mathrm{W}_{\mathrm{intense}}(t, i, s)$. However, a simpler option is to ignore the seeds and only extract the intensity scores from the lexicon.

Definition .10 (Opinion-Aware Total TF Variants)

$$\mathrm{TF}_{\mathrm{total-sentiment-Macro}}(t,d) := \sum_{s\in d}\left(\sum_{i=1}^{n_L(t,s)} \mathrm{W}_{\mathrm{sentiment}}(t,i,s)\right) \quad (124)$$

$$\mathrm{TF}_{\mathrm{total-sentiment-Micro}}(t,d) := \mathrm{tf}(t,d)\cdot \acute{\mathrm{W}}_{\mathrm{sentiment}}(t) \quad (125)$$

$$\mathrm{TF}_{\mathrm{total-intense}}(t,d) := \sum_{s\in d}\left(\sum_{i=1}^{n_L(t,s)} \mathrm{W}_{\mathrm{intense}}(t,i,s)\right) \quad (126)$$

Opinion-Aware Total TF Variants

Opinion-Aware $\mathrm{TF}_{\mathrm{BM25}}$: We need a pivoted term frequency that is built upon sentiments in order to determine the opinion-aware $\mathrm{TF}_{\mathrm{BM25}}$. A list of new pivoted tf variants and their definitions is listed in definition .11. The rationale is as follows. A scaling of the total TF $\mathrm{tf}(t,d)$ is not advisable since this would have implications on the document length. A scaling of the $\mathrm{TF}_{\mathrm{BM25}}$ would just equate to a linear scaling of the TF-IDF weight. The determination of the opinion-aware $\mathrm{TF}_{\mathrm{BM25}}$ is presented in definition 12.

Definition .11 (Pivoted term frequencies)

$$\mathrm{tf}_{\mathrm{piv,sentiment-Macro}}(t,d) := \frac{\mathrm{TF}_{\mathrm{total-sentiment-Macro}}(t,d)}{(k_1\,(b\cdot \mathrm{pivdl}+1-b))} \quad (127)$$

$$\mathrm{tf}_{\mathrm{piv,sentiment-Micro}}(t,d) := \mathrm{tf}_{\mathrm{piv}}(t,d)\cdot \acute{\mathrm{W}}_{\mathrm{sentiment}}(t) \quad (128)$$

$$\mathrm{tf}_{\mathrm{piv,intense}}(t,d) := \frac{\mathrm{TF}_{\mathrm{total-intense}}(t,d)}{(k_1\,(b\cdot \mathrm{pivdl}+1-b))} \quad (129)$$

We are choosing lower-case $\mathrm{tf}_{\mathrm{piv}}$ (similar to tf) for indicating that such quantity is input to upper-case TF, the quantity that is used in the TF-IDF formula.

Definition .12 (Opinion-Aware $\mathrm{TF}_{\mathrm{BM25}}$ Variants.)

$$\mathrm{TF}_{\mathrm{BM25-sentiment-Macro}}(t,d) := \frac{\mathrm{tf}_{\mathrm{piv,sentiment-Macro}}(t,d)}{|\mathrm{tf}_{\mathrm{piv,sentiment-Macro}}(t,d)|+1} \quad (130)$$

$$\mathrm{TF}_{\mathrm{BM25-sentiment-Micro}}(t,d) := \frac{\mathrm{tf}_{\mathrm{piv,sentiment-Micro}}(t,d)}{|\mathrm{tf}_{\mathrm{piv,sentiment-Micro}}(t,d)|+1} \quad (131)$$

$$\mathrm{TF}_{\mathrm{BM25-intense}}(t,d) := \frac{\mathrm{tf}_{\mathrm{piv,intense}}(t,d)}{|\mathrm{tf}_{\mathrm{piv,intense}}(t,d)|+1} \quad (132)$$

For the scientific study of intensity, in this thesis we focused on opinion words. All of the proposed models consider the neutral terms within queries as stop words. However, future studies are needed to explore the integration of topical retrieval with opinion words.

Proposed models

In this section we describe the incorporation of TF variants into the TF-IDF and LM models.

TF-IDF The proposed TF-IDF consists of the well-known IDF as the notion of rareness and opinion-based term frequencies:

$$\mathrm{W}_{\mathrm{TF\text{-}IDF-x}}(t,d) := \mathrm{TF}_{\mathrm{x}}(t,d) \cdot \mathrm{IDF}(t), \quad (133)$$

where x is a generic type which can be any of different forms of opinion-aware approaches including $\mathrm{total}-\mathrm{sentiment}-\mathrm{Macro}$, $\mathrm{total}-\mathrm{sentiment}-\mathrm{Micro}$, $\mathrm{total}-\mathrm{intense}$ and the corresponding BM25 approaches. The Retrieval Status Value (RSV) is the sum of TF-IDF weights across document terms:

$$\mathrm{RSV}_{\mathrm{TF\text{-}IDF-x}}(d,c) := \sum_{t \in d} \mathrm{W}_{\mathrm{TF\text{-}IDF-x}}(t,d). \quad (134)$$

Language Modelling For LM, we need an approach that considers sentiment when estimating the probability $p(t|d)$ and $p(t|c)$, respectively. For reflecting the fact that we apply negative values (because of the polarity), we introduce the notation $\pi(t|d)$ and $\pi(t|c)$. We hire opinion-aware term

frequencies introduced in the previous section and incorporate them into the probabilities. Therefore we determine the new parameters as follows:

$$\pi_{\mathrm{x}}(t|d) := \frac{\mathrm{TF}_{\mathrm{total-x}}(t,d)}{|d_o|} \tag{135}$$

$$\pi_{\mathrm{x}}(t|c) := \frac{\mathrm{TF}_{\mathrm{total-x}}(t,c)}{|c_o|}, \tag{136}$$

where x is the type of opinion-aware model, $|d_o|$ is the opinion-based document length $\sum_{o \in d} n_L(o,d)$ and $|c_o|$ is the opinion-based collection length $\sum_{o \in c} n_L(o,c)$ for sentiment-bearing term o. The determination of the length parameters is dependant on the model-type x.
The calculation of opinion-aware LM would result in issues related to negative values within logarithm. To address this issue, we apply the logarithm to the absolute result of the division and deliver the polarity of term sentiment into the formula by the use of $\mathrm{TF}_{\mathrm{x}}(t,q)$ parameter which determines the sign. In LM, the TF quantification is for the *query*; in other words, on the TF-IDF side of IR, it is more straight-forward to generalise the TF regarding sentiment. The equation below shows the opinion-aware LM:

$$\mathrm{W}_{\mathrm{LM-x}}(t,d,q,c) := \mathrm{TF}_{\mathrm{x}}(t,q) \cdot \log\left(\left|\frac{(1-\sigma_d)\cdot\pi_{\mathrm{x}}(t|c) + \sigma_d \cdot \pi_{\mathrm{x}}(t|d))}{\pi_{\mathrm{x}}(t|c)}\right|\right), \tag{137}$$

The document is ranked by dividing the smoothed version of the multinomial probability of the query given the document by the probability of the query in the collection.

$$\mathrm{RSV}_{\mathrm{LM-x}}(d,q,c) := \sum_{t \in q} \mathrm{W}_{\mathrm{LM-x}}(t,d,q,c) \tag{138}$$

5.2.3 Evaluation

Experiments

The contribution of this section is twofold. First we measure the effectiveness of the proposed sentiment-aware models in the polarity classification task by comparing them with the VADER algorithm. Second, and more importantly, we evaluate our intensity-aware models based on positive and negative query

sets. We compare the results with basic plain IR models and perform a query-based analysis to confirm the high quality of the intensity-based retrieval on single queries.

Evaluation of Sentiment Classification

We evaluated the quality of the sentiment-polarity classification task with regards to the proposed methods. We applied the plain sentiment-based VADER as well as the proposed sentiment-aware IR models on the IMDB dataset containing 25000 highly polar reviews [69]. The dataset is divided equally into negative and positive parts. We used the data and their labels as the gold standard for this task.

Model	P@1000		R@1000		F1	
	pos	*neg*	*pos*	*neg*	*pos*	*neg*
IMDB Reviews						
$\text{sentiment}_{\text{vader}}$	0.893	0.724	0.0714	0.0579	0.132	0.107
$\text{TF-IDF}_{\text{total-sentiment-Macro}}$	0.905	0.764	0.0724	0.0611	0.134 (+1.5%)	0.113 (+5.6%)
$\text{TF-IDF}_{\text{total-sentiment-Micro}}$	0.898	0.762	0.0718	0.0610	0.132 (+0.0%)	0.112 (+4.7%)
$\text{TF-IDF}_{\text{BM25-sentiment-Macro}}$	0.953	0.804	0.0762	0.0643	0.141 (+6.8%)	0.119 (+11.2%)
$\text{TF-IDF}_{\text{BM25-sentiment-Micro}}$	0.943	0.801	0.0754	0.0641	0.139 (+5.3%)	0.118 (+10.3%)
$\text{LM}_{\text{sentiment-Macro}}$	0.926	0.773	0.0741	0.0618	0.137 (+3.8%)	0.114 (+6.5%)
$\text{LM}_{\text{sentiment-Micro}}$	0.919	0.780	0.0735	0.0624	0.136 (+3.0%)	0.115 (+7.5%)

Table 11: Sentiment polarity classification: Sentiment-aware macro TF-IDF is more effective than other models. All new models outperformed the baseline.

To generate Precision@(K=1000) and Recall@(K=1000) results, the top 1000 reviews retrieved by models are labelled as *pos* and accordingly, the top 1000 reviews from the bottom of the reversed result-lists fell into the *neg* class. Due to the absence of time and resources and for the sake of simplicity, I used these heuristic values for variable K. The sentiment-based models are query-independent as they are used for the classification task. The lack of queries and the nature of the classification task led to the use of particular evaluation metrics including Recall@K, Precision@K and F1 score.
Table 11 column 4 lists F1 scores for our runs on the dataset. The data in this column indicates that all of the sentiment-aware models outperformed

the baseline $\text{sentiment}_{\text{vader}}$ concerning both negative and positive classifications. The macro instance of the $\text{TF-IDF}_{\text{BM25}}$ achieved the highest score among the models. The advantage of the macro group is the consideration of influencers in retrieval. As we expected, they provided higher scores than the micro models although the differences are not statistically significant. As can be seen in the last column of table 11 (F1 scores), in contrary with the baseline, the performance of the novel models was higher for the negative class compared to the positive one.

Evaluation of intensity-aware models

The task is to retrieve reviews (documents) with a similar intensity score to the target review (query). I used the IMDB-review collection as the primary dataset to evaluate the performance of the candidate models. For confirming the results, I used 2000 DVD reviews taken from the Amazon Sentiment Dataset [15]. The basic models are term-based, whereas the intensity-aware models only consider the intensity values of sentiment-bearing terms within queries and documents (neutral words are ignored).
Each query set consists of 50 reviews which are correspondent to the query set label in terms of polarity. Therefore, there are 50 positive and 50 negative reviews within each dataset. As an example, *'There are scenes which make you gulp with sudden emotion, and those which even put a smile on your face through ...'* is a snippet of a positive query that we used. Using a labelled review as a query is expected to retrieve the similar reviews concerning the negative and positive sentiment-polarity categorizations.
To evaluate the intensity-aware models and their corresponding basic models, I used Mean Average Precision (MAP) and Reciprocal Rank as shown in table 12. Probably the most interesting observation is that all of the novel models provided higher MAP scores than the basic models for both query sets. The models were more effective for negative reviews than the positive ones concerning IMDB dataset whereas, they provided much higher scores for the positive set compared to the negative queries when applied on Amazon DVD reviews.
$\text{LM}_{\text{intense}}$ and $\text{TF-IDF}_{\text{BM25-intense}}$ achieved the highest MAP and Reciprocal Rank values. Although the intensity-aware LM provided a higher MAP score than the macro version of the $\text{TF-IDF}_{\text{BM25}}$, the variance is extremely small.

Model	MAP				Reciprocal Rank
	pos	*neg*	*avg*		*total*
IMDB Reviews					
TF-IDF (Eq (62))	0.286	0.237	0.261		0.80
$\text{TF-IDF}_{\text{total-intense}}$	0.320	0.345	0.332	(+27.2%)	0.86
$\text{TF-IDF}_{\text{BM25}}$ (Eq (5))	0.293	0.234	0.263		0.87
$\text{TF-IDF}_{\text{BM25-intense}}$	0.351	0.322	<u>0.336</u>	(+27.76%)	<u>1.00</u>
LM (Eq (25))	0.259	0.241	0.250		0.60
$\text{LM}_{\text{intense}}$	0.305	0.369	<u>0.337</u>	(+34.8%)	<u>1.00</u>
Amazon DVD Reviews					
$\text{TF-IDF}_{\text{total-basic}}$	0.288	0.111	0.199		0.826
$\text{TF-IDF}_{\text{total-intense}}$	0.478	0.115	0.296	(+48.74%)	0.895
$\text{TF-IDF}_{\text{BM25-basic}}$	0.280	0.125	0.202		0.886
$\text{TF-IDF}_{\text{BM25-intense}}$	0.471	0.138	<u>0.304</u>	(+50.49%)	<u>1.000</u>
LM_{basic}	0.281	0.090	0.185		0.750
$\text{LM}_{\text{intense}}$	0.465	0.145	<u>0.305</u>	(+64.86%)	<u>0.995</u>

Table 12: Evaluation of Intensity-Aware Retrieval Models: Intense models had higher F1 scores than the baseline. The $\text{TF-IDF}_{\text{BM25}}$ intense model had the highest score.

Query-based Analysis

To further investigate the features which affect the performance of our approach, we performed a query-based analysis on the evaluation results captured from IMDB dataset. This analysis resulted in the determination of the number of queries which are more compatible with the novel models for both positive and negative query sets. The intuitive idea is that a model might have a high MAP while the distribution of it's Average Precision (AP) scores is not acceptable across the queries.

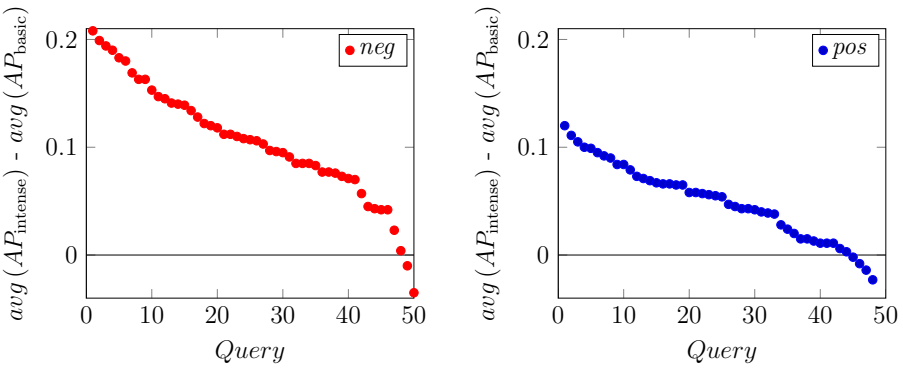


Figure 13: Distribution of Avg-AP differences between intense and basic models considering 100 queries (in descending order): Query analysis shows intense models were more effective for roughly 96% of queries compared to the basic models.

I calculated the average of APs for both basic and intensity-aware models separately and subsequently for each query set, plotted the distribution of the differences ordered by descending which can be seen in Figure 13. As the figure shows, the intensity aware models were more effective on more than 95% of the queries compared to the basic approach.
The average AP difference of the models is an indication of the quality of the novel approach over the basic models on a query basis. Figure 14 shows the positive correlation between the quality of the intensity-aware models and the ratio of the query intensity to the query length. Interestingly, the correlation is stronger for positive queries which shows that the polarity of the queries could impact effectiveness of the models. The Pearson correlation coefficient for positive query set is 0.21 and for the negative query set is estimated as weak positive. This suggests that further experiments need

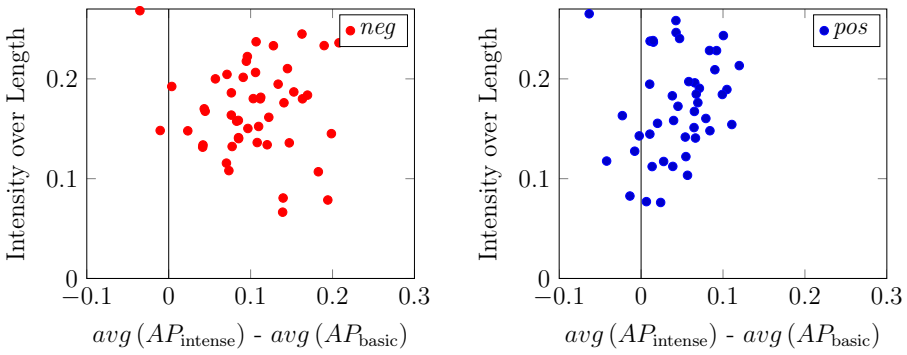


Figure 14: Pearson correlation between Avg-AP differences and the ratio of query intensity to query length: The correlation value for positive queries is 0.21 while the relationship between the parameters is weak positive regarding negative queries.

to be conducted to discover additional features that resulted in the higher quality of the intensity-aware approach.

5.2.4 Discussion

In this section, I presented two novel families of opinion-aware models, namely sentiment-aware and intensity-aware models. The sentiment-aware models are developed to enhance the effectiveness of sentiment classification, whereas the intensity-aware models are retrieval methods. The main purpose of such an approach was to deal with the problem of opinion words with low document frequency and high intensity.

The contribution of the section was twofold. Firstly, I compared the sentiment-aware models (for the classification task) with the VADER rule-based algorithm and confirmed that the consideration of a notion of IDF as used in my models improves the VADER sentiment classification.

Secondly, in another task (similar review retrieval), I applied basic (term-only) and intensity-aware retrieval models to movie reviews and tested to find out if items of a specific polarity retrieve similar items. All of the intensity-aware models outperformed their corresponding basic models and it turned out that the effectiveness of the novel models is consistent across a range of positive and negative queries. Additionally, this study found a positive correlation between the performance of novel models and the ratio of query-

intensity to query-length.
We hope this section facilitates the establishment of clear standards to be used by companies and organizations that need to track customer's overall feelings towards products.

Future work

It is important to improve opinion-aware models by quantifying the IDF variant based on opinions. By quantifying IDF for frequent opinions with high intensity, I aim to make the models more intelligent. The table below shows a distribution sample of the document and collection frequencies concerning opinions in the OHSUMED dataset:

Term	**Document Frequency**	**Collection Frequency**	**Intensity Score**
good	6162	7067	1.9
hard	429	573	-0.4
hero	2	3	2.6
weak	1202	1363	-1.9
effectively	1971	2068	1.9
popular	356	384	1.8
poor	5296	6420	-2.1

Table 13: Example of the statistics captured from OHSUMED containing document frequency, collection frequency and sentiment score of the terms.

As can be seen, although terms such as *effectively*, *weak* and *poor* are high in intensity, they appear many times in the collection and have high document frequencies too. Considering the important role of opinion terms in retrieval, we aim to come up with new IDF weights. Below I list some IDF candidates which take into consideration opinion terms.

- **IDF based on intensity frequency**: The most interesting IDF candidate is the result of leveraging the intensity frequency score. This

variant is simply the number of documents containing any lexical feature with an intensity range of the one for the current term.

For example, let the intensity score of the term *good* be 2.4 and its document frequency be 2500, we count the number of documents in which an intensity range of (2.2-2.4) could be observed. To accomplish this task, we need to index all possible ranges of intensity frequencies in the collection. The table below is an example representation of frequencies based on intensity ranges:

Range	**Frequency in collection**
(0,0.2)	250
(0.2,0.4)	500
(0.4,0.6)	129
(2.2,2.4)	125
(1.0,1.2)	630

Table 14: Example of the intensity-frequency representation.

let δ be the intensity-frequency of a term, the IDF candidate is defined as follows:

$$\text{IDF}_{\text{i-range}}(\delta) := \log\left(\frac{N}{\delta}\right) \tag{139}$$

- **IDF based on sigmoid function**: This approach tunes the intensity score of a term to be between 0 and 1 and then multiplies it by the traditional IDF weight. The result weight is influenced by two factors; rareness and intensity. The intensity can be normalized using the sigmoid function. Let n be a real number, the intensity factor is defined

as follows:

$$\mathrm{IF}\,(t) \;:=\; \frac{e^{(n\cdot \mathrm{W}_{\mathrm{intense}}(t))}}{1+e^{(n\cdot \mathrm{W}_{\mathrm{intense}}(t))}} \tag{140}$$

Finally, the IDF candidate is:

$$\mathrm{e_n}\,(t,c) := \mathrm{IF}\cdot \mathrm{IDF}\,(t,c) \tag{141}$$

- **IDF based on regulation of document frequency**:

$$\mathrm{df}_{\mathrm{intense}}\,(t) := \frac{\mathrm{df}}{\mathrm{W}_{\mathrm{intense}}\,(t)} \tag{142}$$

In the above equation, we decrease the document frequency weight for the terms with a higher intensity. For example, let the document frequency of the term *good* be 2500 and its intensity score be 2.4, the new intense-aware document frequency is the division of 2500 by 2.4. The weight has a proportionally higher IDF which could resolve the problem with high frequency in regards to opinion terms. Consequently, the IDF based on the regulation of the parameter df is as follows:

$$\mathrm{IDF}_{\mathrm{df}_{\mathrm{intense}}}\,(t) := \log\left(\frac{N}{\mathrm{df}_{\mathrm{intense}}\,(t)}\right) \tag{143}$$

Chapter 6

IR-based Recommendation and Notification

6.1 Chapter overview

This chapter aims at showing how advanced IR could be transferred to the urgent notification filtering. In chapter 4, I justified the effectiveness of DCM-based IR (FDCM) in semantic retrieval and chapter 5 confirmed the capability of applying IR to the opinion-aware retrieval. Subsequently, this chapter proceeds to proposing a recommender instance of the FDCM framework to be utilized in the notification filtering task (section 6.2). Additionally, we publish a novel benchmark specifically for evaluating IR on semantically enriched texts in order to add to evidence that semantic IR is a robust solution for urgent notification filtering (section 6.3).
Models become complex and computationally expensive. Therefore, it is desirable to establish theoretically sound, generalizable and thus reusable approaches. We know that KNN-based recommendation has IR roots, is transparent, popular and effective. However, poor retrieval models for the K nearest neighbour, and too simplistic voting (predict the item most popular among the retrieved users) are downsides of KNN. We aim to adapt advanced achievements of semantic-IR to the task of top-N recommendation through casting semantics to users and items. Section 6.2 is the starting point for building generalizable hybrid recommendation tools in which individual scores of semantic dimensions such as terms, concepts and ratings could be effectively integrated. This could help data scientists to leverage in-

teractions in semantic standards and improve the accuracy of neighbourhood determination. We explore the application of word burstiness and term dependency in collaborative filtering by evaluating the performance of FDCM. We also establish improved-KNN standards based on advanced IR technologies (DCM in this study). The rationale is that DCM is potentially more effective than traditional KNN for finding nearest neighbours and there is no need to separate IR from KNN. Although relevant works show that IR, in general is associated with improvements of collaborative filtering, we need a more in-depth investigation of the association between collaborative filtering and IR-based features (e.g. analysis of user-based rankings, dependency assumptions and query performance predictions). I use some of these features for training a recommender system in two settings: in the first setting, rating is equivalent to item-frequency of IR whereas, in the second setting, play count (user interaction is not limited to a fixed range of ordinal numbers) is the frequency. Section 6.2 introduces two models for top-N recommendation: $\text{RFDCM}_{\text{Basic}}$ and RFDCM_{LR}. The first model, $\text{RFDCM}_{\text{Basic}}$ improves traditional KNN by using a DCM variant as the similarity measure. The second model, RFDCM_{LR} effectively combines IR-inspired statistics with Logistic Regression (LR). The section shows that the combination of IR and ML could enhance the accuracy of recommendation. The experimental study (carried out on MovieLens, Book Crossing and LastFM datasets) shows that the retrieval-inspired algorithm outperforms recommendation baselines such as KNN (with mean-squared distance) and matrix factorisation. Overall, section 6.2 encourages data scientists to employ IR-inspired algorithms for recommendation tasks.

Researchers need a benchmark which primarily takes into consideration the integration of opinions and medical concepts. This is due to the importance of feelings in detecting the level of urgency in medical domain (we need explanations of IR features for urgent notification filtering task). Moreover, bio-medical companies need to analyse customer's general feelings about their products. On the other hand, patients need to know the sentiment of product reviews before buying. Wherefore the examination of sentiments would be beneficial for both buyers and suppliers of medical products.

In section 6.3, we address this problem by creating and making available a medical benchmark specifically for the task of opinion-aware retrieval.

Bio-medical benchmarks consider various pillars of semantics in collections and queries, e.g., terms, concepts and attributes. These semantics would enable data scientists to develop effective models for different tasks, e.g.,

filtering and classification. In section 6.3, we create a benchmark that consists of a dataset, a query-set and the relevance results. The dataset consists of Amazon reviews for medical products. Additionally, it supports the use of common semantics (terms, concepts, opinions and relations) in biomedical retrieval.
The second contribution of the section is to apply sentiment-aware models to the dataset. We propose a family of opinion-aware models for ranking medical reviews. By consolidating the methods for modelling opinions and sentiments in medical ranking, we aim to address the deficiencies in different tasks including but not limited to notification filtering and review filtering. Section 6.3 contributes to improving medical review filtering through IR. It is the starting point of developing models that could better meet the needs of bio-medical organizations, companies and individual buyers for analysing most critical, positive and negative reviews.

6.2 RFDCM: Recommender system based on FDCM

6.2.1 Background

Collaborative filtering recommends personalized items to users based on past user interactions including preferences, opinions and ratings [44]. Recommendation algorithms could be classified in three categories: matrix factorization, KNN and neural networks. In this section we add semantic IR to this classification by exploring the usage of FDCM framework in recommendation. Firstly, we introduce the extension of FDCM for ranking similar users and secondly, demonstrate two approaches for recommending items by using features of the model. Specifically, this section incorporates a semantic retrieval framework and the Dirichlet-multinomial distribution into the recommendation task. The motivation underlying this approach is that the consideration of semantic information and the properties of the Dirichlet-Compound-Multinomial (DCM) is known to be beneficial for document retrieval, and therefore, could be conducive to recommendation as well. Current recommender systems apply traditional information retrieval (IR) for reasoning about similarity (e.g. cosine similarity or more advanced models) between users and between items. Modern recommender systems will apply advanced similarity algorithms for training the recommender model.

There have been advancements in IR regarding the usage of semantics and probabilistic models such as the DCM. Consolidating advanced IR methods for applying them in the context of collaborative filtering, is gaining attention [105]. However, the effects of advanced IR concepts (e.g. DCM, burstiness and semantics) in collaborative filtering are not well studied. As discussed in chapter 3, several works improved the original language modelling (LM) [84] such as Dirichlet Compound Multinomial (DCM). This model is the result of leveraging a well-defined probability distribution and a Polyà urn process in which burstiness is strongly taken into consideration [28]. In this section, we study the application of FDCM (introduced in section 4.4 of chapter 4) in collaborative filtering and tune its recommender-based extensions. Furthermore, we present a recommender approach derived from incorporating statistics of FDCM into Logistic Regression (LR). We apply Recommender FDCM (RFDCM) to MovieLens dataset where user interaction is rating and restricted to a fixed range of ordinal numbers. Additionally, we apply it to Book Crossing and LastFM data and analyse the results obtained from datasets with different types of user interactions. In this thesis, we also examine the use of FDCM as a measure of user similarity in classic KNN.

Recommendation-oriented FDCM (RFDCM)

We introduce a mapping between recommendation and retrieval. The mapping helps to transfer IR concepts (documents and queries, terms, term frequency) to the task of Collaborative Filtering with regards to user interactions (rating and play count) data. Table 15 shows the mapping and explains how we bind user interactions to IR concepts.
A user corresponds to a query constructed from a partial vector of item ratings. In this scenario, the information needs (answers) are high ratings of user which are absent in the query. The query is formulated as all items of user along with relevant frequencies (user interactions) except for answers.
When item rating is assumed to be the term frequency, queries become long. This means the average within-query term frequency is high, whereas in traditional IR with short queries, the query term frequency is rarely greater than 1. To rank the users, we use a variant of the FDCM model proposed in [8]. The framework uses an aggregation parameter inspired by IDF to balance semantic weights. The two semantic pillars of the URFDCM are user interactions and concepts. In this section we firstly introduce item-based and concept-based DCMs and then show how we aggregate their ranking scores.

Recommendation	Retrieval
user: set of (item,rating) pairs	Document
item: e.g movies, songs, products	term
subset of user: (item,rating) pairs	query
user interaction: e.g rating, play count	term frequency

Table 15: Mapping to transfer recommendation to retrieval.

6.2.2 Contribution

Item-based DCM (IDCM)

IDCM is for RFDCM while TDCM is for the FDCM framework in section 4.4.1. IDCM considers items (e.g. movies, songs) as terms and the item rating (e.g. for movies the number of stars, for songs the number of views) is considered as the "term frequency". Formally, let q be a query, $\mathrm{IF}(i,q)$ the item frequency (item rating), and c the collection. The IDCM weight and RSV are as follows:

$$\mathrm{W}_{\mathrm{IDCM}}(i,d,q,c) := \mathrm{IF}(i,q) \cdot \log\left((1-\lambda_d) + \lambda_d \cdot \frac{\alpha_d(i)}{\alpha_c(i)}\right) \tag{144}$$

Definition .13 (Item-only-only DCM (IDCM))

$$\mathrm{RSV}_{\mathrm{IDCM}}(d,q,c) := \sum_{i \in d} w_{\mathrm{IDCM}}(i,d,q,c) \tag{145}$$

Eq (145) corresponds to eq (98) (TDCM) for the semantic case.
Regarding the mixture (smoothing) parameter λ_d of the multinomial language model, we used the parameter (Dirichlet prior) discussed for TDCM in section 4.4.1 ([119]).

User-based Recommender Full DCM (URFDCM)

In this section, we define the user-ranking model constructed from the aggregation of item-based and concept-based instances of FDCM. The concept-based FDCM is shown in definition (7). With regards to the recommendation scenario, concepts are vectors of tags attached to the user (e.g. genre of a movie). We name the model User-based Recommender Full Dirichlet Compound Model or URFDCM. To make the formulations readable, the type-aware X indicator is hired with function names, e.g. $\mathrm{TF}(i)$ is the item frequency of item i in the document, whereas $\mathrm{CF}(\varphi)$ is the frequency of concept φ in document. The generic FDCM framework could help to easily incorporate other semantic predicates such as sentiments or relationships. Let the parameter σ_I be the linear aggregation parameter, the URFDCM is defined as follows:

Definition .14 (User-based RFDCM (URFDCM))

$$\begin{aligned}&\mathrm{RSV}_{\mathrm{URFDCM}}(d,q,c) :=\\ &\quad \sigma_I \cdot \mathrm{RSV}_{\mathrm{IDCM}}(d,q,c) + (1-\sigma_I)\cdot \mathrm{RSV}_{\mathrm{CDCM}}(d,q,c)\end{aligned} \tag{146}$$

Note the symmetry between eq (146) and eq (102) (FDCM): For the URFDCM score, we combine IDCM (item-based score) with CDCM (concept-based score); for the FDCM score, we combined TDCM (term-based score) with CDCM.

Item-Concept Aggregation Parameter σ_I

The sum of IDFs has been used for the term-concept aggregation parameter σ_T (section 4.4.2) in the FDCM model. However in the recommendation scenario, we find item-based IDF unbalanced with concept-based IDF. Therefore, we employ the max of IDFs as it is a more sensible choice. The difference between the maximum of item-based IDFs and the maximum of concept-based IDFs leads to the definition of the recommender semantic information denoted $\hat{\mathrm{q}}_{\text{sem-info}}$:

$$\hat{\mathrm{q}}_{\text{sem-info}} := \left(\max_{\varphi\in q} \mathrm{IDF}(\varphi)\right) - \left(\max_{i\in q} \mathrm{IDF}(i)\right) \tag{147}$$

The semantic information in eq (147) is based on the maximum IDFs, whereas the semantic information for FDCM in eq (104) was based on the sum of IDFs.

As for FDCM (and again following [78]), there are two steps for transforming the semantic information to the linear aggregation parameter σ_I. The first step is the frequency saturation:

$$\hat{q}_{\text{sf}} := \frac{\hat{\text{q}}_{\text{sem-info}}}{1 + |\,\hat{\text{q}}_{\text{sem-info}}\,|} \tag{148}$$

$\hat{q}_{\text{sf}}$ in eq (148) is for recommendation-based FDCM (RFDCM) what qsf in (105) is for FDCM. The next step combines the item-based query-length $|\,q\,|$ using $\hat{q}_{\text{sf}}$:

$$\hat{q}_{\text{raw-sem-length}} := \hat{q}_{\text{sf}} \cdot |\,q\,| \tag{149}$$

$\hat{q}_{\text{raw-sem-length}}$ is for recommendation-based FDCM (RFDCM) what $\text{q}_{\text{raw-sem-length}}$ in (107) is for FDCM.
The final step is a logarithmic transformation inspired by Paik [78], and in more general, by logarithmic discounting functions such as discounted cumulative gain (DCG). This leads to the aggregation parameter σ_I which is corresponding to σ_T in eq (112):

$$\sigma_I := \frac{1}{\log_2\left(1 + \hat{q}_{\text{raw-sem-length}}\right)} \tag{150}$$

RFDCM: Basic Variant

The basic approach is KNN as discussed in [2]. Subsequently, we use the below formula to calculate the similarity score of item i given user u:

$$\text{Sim}\,(u, i, M) := \frac{\sum_{v \in M} \text{sim}(u, v) \cdot \text{rating}(v, i)}{\sum_{v \in M} |\,\text{sim}(u, v)\,|} \tag{151}$$

where M is the list of users in the neighbourhood cluster retrieved by URFDCM, $\text{rating}(v, i)$ is the frequency of user interaction (rating or play count) of user v given item i, and $\text{sim}(u, v)$ is the similarity score of the users.
The next step is to exploit the above algorithm in RFDCM, therefore we replace $\text{sim}(u, v)$ with the URFDCM retrieval score from eq (146). Also, $\text{rating}(v, i) := n(i, v)$ which means that the rating is considered as the within-document (within-user) term (item) frequency. Thus, we obtain the following IR-oriented formulation of eq (152):

$$\text{RFDCM}_{\text{Basic}}\,(d, i, c, M) := \frac{\sum_{q \in M} \text{RSV}_{\text{URFDCM}}\,(d, q, c) \cdot n(i, q)}{\sum_{q \in M} |\,\text{RSV}_{\text{URFDCM}}\,(d, q, c)\,|} \tag{152}$$

Essentially, RFDCM aggregates and normalises the URFDCM scores, where each URFDCM score corresponds to the neighbour similarity between user d and user q, and user q has viewed item i $n(i,q)$ times (has given the rating $n(i,q)$).

RFDCM with Logistic Regression (LR)

In this section, I introduce and reason the new features I extracted from URFDCM to train a recommender approach (the second RFDCM variant). This model transforms query and document pairs (train and test sets) to probabilistic features, and it injects these features into an ML model (Logistic Regression in this paper) which ultimately predicts ratings. To keep the focus of this thesis, we used LR as our candidate ML model. This is because LR is shown to perform better compared to some other ML candidates including MLP, SVM and NB (based on our experimental results shown in table 18). However, future work can study the use of more advanced ML approaches for the task.
Each feature is the result of the function $f_x\left(M,i,q\right)$, where M is the ranked-list of top K similar users, i is the input item (e.g. a movie), and q is query. Let $\mu(u,i)$ be the list of features, $\mu_j \in \mu(u,i)$ be a feature at index j, and β_j be the coefficient for the constant, the retrieval score is defined as follows:

Definition .15 (RFDCM with Logistic Regression)

$$\mathrm{RFDCM_{LR}}\left(u,i\right) := \left(1+\exp\left(-(\beta_0+\sum_{j=1}^{n(\mu(u,i))}\beta_j\cdot\mu_j)\right)\right)^{-1} \tag{153}$$

Below we list features in $\mu(u,i)$ along with their descriptions (features are selected based on the correlation analysis):

- **Concept-based Natural Harmony (CNH)** is the harmonic sum (series) concerning concepts associated with item i against query q. Let $n\left(\varphi,q\right)$ be the within-query concept frequency, the CNH quantification is defined as follows:

$$\mathrm{CNH}\left(i,q\right) := \sum_{\varphi\in i}\left(\sum_{g\in q}\left(1+\frac{1}{2}+\frac{1}{n\left(\varphi,q\right)}\right)\right) \tag{154}$$

This gives us a score based on the distribution of concepts (e.g. genres of movies) in query q.

- **Avg Rating (AR)** is the average rating of the item i in M. Let $r\,(i,d)$ be the rating of the input item in document d and $|\,M\,|$ be the number of documents in M, we define AR as $\sum_{d\in M}\left(\frac{r(i,d)}{|\,M\,|}\right)$.

- **Elite Average Rating (EVR)** is the average rating of the input item i in top $\acute{K}$ users of M: $\sum_{d\in \acute{M}}\left(\frac{r(i,d)}{K}\right)$, where $\acute{M}$ is the list of top $\acute{K}$ users (in this paper we used $\acute{K}$=5) .

- **Sum of Total Item Frequency (TIF)** is the sum of total item frequency scores in M. We define the total item frequency as $\sum_{d\in M}\frac{n(i,d)}{\sum_{\acute{i}\in d} n(\acute{i},d)}$.

- **Sum of Total Item Frequency Plus (TIF+)** is the sum of TIF values for users with a rating (for input item i) bigger than variable R : $\sum_{d\in M, r(i,d)>R}\left(\frac{n(i,d)}{\sum_{\acute{i}\in d} n(\acute{i},d)}\right)$.

- **Elite Document Frequency (EDF)** is the total number of users in M who had a rating of $\acute{R}$ or bigger for i: $\mathrm{df}_{d\in M, r(i,d)>\acute{R}}$.

6.2.3 Evaluation

Experimental setup

Data We selected three collections for evaluation, MovieLens, Book Crossing and LastFM. LastFM is a music-artist listening dataset from a set of 1892 users, 92800 play counts and 17632 artists collected from Last.fm music platform [19]. Book Crossing was collected by Cai-Nicolas Ziegler in a 4-week crawl (August-September 2004) [122]. It consists of 271380 books and 4148337 ratings by 95095 users (after data cleaning). Additionally, we used different subsets of MovieLens namely MovieLens-98, MovieLens-2018 and ML-1M in our experiments [40]. Regarding MovieLens, we selected ML-1m as our train-set, while MovieLens-98 and MovieLens-2018 were considered for testing. MovieLens datasets are popular for the task of collaborative filtering on movie ratings. They are collected by the Grouplens Research project at the University of Minnesota. MovieLens-2018 consists of 100836 ratings and

9742 movies across 610 users. The dataset was gathered in September 2018. MovieLens-98 contains 100000 ratings on 1682 movies by 943 users. ML-1m is a collection of 1,000,209 anonymous ratings of approximately 3,900 movies made by 6,040 MovieLens users who joined in 2000. Movie ratings are ordinal values from 1 (very bad) to 5 (very good). The minimum rating means that user strongly dislikes the movie, the maximum score indicates a strong like. Ratings of Book Crossing dataset are between 0 and 10, however, I have added 1 to each rating r, $(r := r + 1)$ to convert the range into (1,11) so that the rating becomes acceptable to be used for term frequency. Concerning LastFM, there is no user-item rating data and therefore, we considered play count as user interaction (term frequency). Unlike MovieLens and Book Crossing ratings, interactions are not restricted to the fixed range (between 1 to 5 and between 1 to 11). When songs of an artist are played frequently, that means user likes them and the corresponding item frequency is high. Table 16 lists statistics over user interaction on LastFM and MovieLens.

	LastFM	**MovieLens-98**	**MovieLens-2018**
#Minimum Frequency	1	0.5	0.5
#25% Percentile	107	3	3
#Median	260	4	3.5
#75% Percentile	614	4	4
#Maximum	352,698	5	5
#Mean	745.2	3.543	3.501
#Std. Deviation	3,751.3	1.058	1.04
#Std. Error of Mean	12.3	0.003	0.003
#Lower 95% CI of mean	721.112	3.537	3.495
#Upper 95% CI of mean	769.375	3.550	3.507

Table 16: User interaction statistics in LastFM and MovieLens.

The interactions on LastFM are more distributed than MovieLens. The minimum play count is 1, the maximum is 352,698, the mean is 745 and the median is 260. Table 17 shows the statistics of the datasets including the task (train or test), number of ratings, users, items and training samples. Additionally, IR-related features of the datasets including the number of queries, number of documents, item-based collection length, distinct item-based collection length, concept-based collection length and distinct concept-based collection length are listed. Furthermore, IR-based probabilistic statistics of the datasets are provided in the table.

Dataset	Task	Entities	Ratings	Users	Items	Train
MovieLens-98	Test	Movies	100000	943	1682	3359
MovieLens-2018	Test	Movies	100836	610	9742	3597
MovieLens-1m	Train	Movies	1000209	6040	3900	-
Book Crossing	Test+Train	Books	4148337	95095	271380	5381
LastFM	Test+Train	Artists	92800	1892	17632	4916

Dataset	Queries	Documents	$\|c_i\|$	$\|\overline{c_i}\|$	$\|c_\varphi\|$	$\|\overline{c_\varphi}\|$
MovieLens-98	50	671	342682	99079	907535	11160
MovieLens-2018	50	610	339152	99643	923792	10154
Book Crossing	200	95095	4148337	1041926	-	-
LastFM	100	1892	69185867	92834	-	-

Table 17: IR-based cardinality of datasets.

Implementation and Reproducibility The test collections used in the section are available online. For making this work reproducible, we outline technical details and assumptions considered in the study. We iterated through user-item vectors of datasets using data-frames of *python* and accordingly created the documents. To build the query set, we followed two steps. Firstly, using the *train-test-split* function provided by the surprise package [52], random-state=0 and test-size=0.25, we sampled the documents. Secondly, we applied random sampling without replacement onto the captured *test-set* and filtered out test-set (queries) from each dataset. Throughout the user-ranking process, we utilised the Lucene framework with Dirichlet Prior to index the documents (generated in the previous step) and retrieved the top 50 pseudo relevant documents for each query. The results were then re-ranked by the RFDCM models. To determine the Dirichlet parameters of DCM, we considered $\mu = 2000$ for $\lambda_d := \mu/(\mu + |d|)$ in eq (144) (page 114). Additionally, 15 iterations were used for estimating parameter m_c of eq (96) in page (79). Also, we generated the so-called IR-based samples for every single user-document pair. Concerning variables R and $\acute{R}$ of TIF+ and EDF, we used (R=3.5 and $\acute{R}$=4.5 for MovieLens) and (R=200 and $\acute{R}$=500 for LastFM). We considered high values for $\acute{R}$ in order to take into consideration users who extremely enjoyed input-item i. Regarding the train-set generation for MovieLens, we used 3597 samples based on 100 queries gathered from MovieLens-1m whereas 4916 training samples were used for LastFM (extracted from 400 queries of the same dataset). Furthermore, 5381 ratings used were utilized for training Book Crossing over 200 queries. The thresholds δ and Δ were considered to assess the relevance of items (item is relevant only if the rating is bigger than the threshold). We used the thresholds $\delta = 3.5$ and $\Delta = 4.0$ in MovieLens, ($\delta = 3.0$ and $\Delta = 5.0$) in Book Crossing and ($\delta = 50.0$ and $\Delta = 200.0$) in LastFM. Finally, we fed the training data into the *sklearn* [81] implantation of Logistic Regression (solver='lbfgs', multi class='ovr' and random-state=0) and performed top=N recommendation. All other ML candidates in the thesis are based on *sklearn* using default parameters. The built-in *predict-proba* function of ML models were used for ranking top-N items.

Baselines We exploited the implementations of the surprise library [52] and used its default parameters for the baselines.

- **KNN baselines** includes three models namely $\text{KNN}_{\text{Basic}}$, KNN_{Mean}

and $\text{KNN}_{\text{Zscore}}$. Both user-based and item-based settings are considered in each model. Mean Squared Difference (MSD) distance was applied in all variants as the measure of similarity. $\text{KNN}_{\text{Basic}}$ is the core KNN algorithm of the surprise library [52]. KNN_{Mean} is a reinforced variant of the basic KNN model in which the mean ratings of each user is taken into account. $\text{KNN}_{\text{Zscore}}$ is the result of integrating z-score normalization of each user with the basic approach. In our experiments we used the default parameter values $k = 40$ and $mink = 1$ (minimum number of neighbors for aggregation)

- **Matrix Factorization baselines** include SVD [98] and Non negativistic Matrix Factorization (NMF) [112]. SVD is a substantial approach in which matrix is based on singular value decomposition.

- **Slope-One** is a quite simpler collaborative filtering based on the average deviation between similar groups of users [60]. It takes into consideration the average differences of ratings for items that have been rated by user.

- **Co-Clustering** involves simultaneous clustering of users and items [34] which is developed based on generalized maximum entropy [9].

URFDCM: Analysis

To assess the effectiveness of URFDCM, I analysed the statistical properties of the training samples of the datasets against the statistics of items to be recommended. As mentioned earlier, each sample consists of IR-inspired features derived from URFDCM results. Table 19 illustrates this analysis and shows statistics including Reciprocal Rank distribution (which helps to understand the performance of the model in retrieving very top items) for relevant items.
Reciprocal Rank (RecipR) is defined as dividing 1 by URFDM rank of the top-one user who interacted with input-item. The results on MovieLens dataset turned out to be very promising since 71% of RecipR values were 1, while only less than 1 percent of the samples had a RecipR smaller than 20%. Although MovieLens achieved better results than LastFM, the number of LastFM samples with a Reciprocal Rank of 1 is still considerable (25%). This analysis confirms the strong similarity between top-1 user determined by URFDM and the target user. The table also shows high values for EDF

Candidate	**MAP@5**	**MAP@10**	**HR@5**	**HR@10**
threshold δ (50)				
MLP	0.277	0.435	0.403	0.741
SVM	0.167	0.389	0.332	0.769
NB	0.271	0.431	0.375	0.728
LR	**0.420***	**0.553***	**0.550***	**0.794**
threshold Δ (200)				
MLP	0.280	0.440	0.421	0.725
SVM	0.133	0.364	0.299	0.757
NB	0.240	0.403	0.350	0.721
LR	**0.389***	**0.545***	**0.542***	**0.803***

Table 18: Evaluation of ML candidates for RFDCM in LastFM: Logistic Regression (LR) is more effective than ML baselines. Super-Script (*) denotes the statistical significance improvement against baselines using the paired t-test with p-value <0.05.

	MovieLens	LastFM
#(%)Receipt.Rank=1	71	25
#(%)Receipt.Rank=0.5	12	11
#(%)Receipt.Rank=0.33	4	5
#avg.EDF	14.86	6.82
#avg.EVR	3.77	846.62

Table 19: Analysis of URFDCM performance: Samples with a significantly small Reciprocal Rank construct a considerable proportion of the dataset. The distribution of EDF scores shows strong relationship between RFDCM and similarity of users in terms of interest in the same products.

concerning both datasets. The average EDF score in MovieLens is 14.68 whereas, this value is 6.82 in LastFM. Moreover, the Average Rating of item i in top K results (EVR) is 3.77 for MovieLens which is 846 for LastFM.

Analysis of RFDCM results

Evaluation of ML candidates Table 18 shows the results for RFDCM_{LR} and ML baselines. The measures used were Mean Average Precision at 5 (MAP@5), MAP@10, Hit Rate at 5 (HR@5) and HR@10. The baselines are MLP (Multi-Layer Perceptual), SVM (Support Vector Machine) and NB (Naive Bayes with multinomial distribution). For assessing the relevance, we used thresholds δ and Δ (δ means that user interactions must be positive while Δ is for strictly positive interactions).

Evaluation of RFDCM Experimental results for comparing RFDCM_{LR} and $\text{RFDCM}_{\text{Basic}}$ are listed in table 20 (MovieLens results) and table 21 (Book Crossing and LastFM results). The tables listed the results in two settings (threshold $> \delta$ and threshold $> \Delta$). The evaluation metrics for this task were MAP@5, MAP@10 and HR@10. The main finding is that in most cases RFDCM_{LR} improved the baselines. Apart from one exception in each setting, it outperformed baselines derived from matrix factorization and KNN on all

Model	MovieLens-98 ($\delta = 3.5$, $\Delta = 4.0$)			Mons-2018 ($\delta = 3.5$, $\Delta = 4.0$)		
	MAP@5	**MAP@10**	**HR@10**	**MAP@5**	**MAP@10**	**HR@10**
$\mathbf{KNN_{Basic+[Item\text{-}based]}} > \delta$	0.162	0.243	0.389	0.134	0.222	0.352
$\mathbf{KNN_{Basic+[User\text{-}based]}} > \delta$	0.177	0.262	0.406	0.183	0.256	0.385
$\mathbf{KNN_{Means+[Item\text{-}based]}} > \delta$	0.160	0.254	0.423	0.144	0.220	0.342
$\mathbf{KNN_{Means+[User\text{-}based]}} > \delta$	0.146	0.240	0.393	0.190	0.260	0.342
$\mathbf{KNN_{ZScore+[Item\text{-}based]}} > \delta$	0.159	0.251	0.42	0.141	0.211	0.342
$\mathbf{KNN_{ZScore+[User\text{-}based]}} > \delta$	0.154	0.237	0.383	0.164	0.244	0.338
Co-Clustering $> \delta$	0.152	0.263	0.416	0.133	0.202	0.323
SVD $> \delta$	0.135	0.248	0.426	0.108	0.227	0.385
SlopeOne $> \delta$	0.163	0.257	0.410	0.128	0.215	0.366
NMF $> \delta$	0.168	0.237	0.376	0.137	0.229	0.390
$\text{RFDCM}_{Basic} > \delta$	0.107	0.189	0.339	0.113	0.186	0.338
$\text{RFDCM}_{LR} > \delta$	0.174	**0.277**	**0.450**	$\mathbf{0.216}^{\theta\zeta\alpha}$	$\mathbf{0.276}^{\theta}$	$\mathbf{0.404}^{\theta}$
% ***better than SVD***	+28.89	+11.69	+5.63	+100	+21.58	+4.93
$\mathbf{KNN_{Basic+[Item\text{-}based]}} > \Delta$	0.133	0.191	0.266	0.074	0.165	0.311
$\mathbf{KNN_{Basic+[User\text{-}based]}} > \Delta$	0.172	0.214	0.283	0.074	0.150	0.300
$\mathbf{KNN_{Means+[Item\text{-}based]}} > \Delta$	0.139	0.194	0.300	0.088	0.141	0.288
$\mathbf{KNN_{Means+[User\text{-}based]}} > \Delta$	0.148	0.196	0.258	0.081	0.134	0.233
$\mathbf{KNN_{ZScore+[Item\text{-}based]}} > \Delta$	0.142	0.184	0.266	0.098	0.139	0.255
$\mathbf{KNN_{ZScore+[User\text{-}based]}} > \Delta$	0.148	0.189	0.250	0.098	0.139	0.233
Co-Clustering $> \Delta$	0.159	0.218	0.316	0.0888	0.0885	0.200
SVD $> \Delta$	0.137	0.225	0.308	0.105	**0.218**	0.300
SlopeOne $> \Delta$	0.142	0.215	0.308	0.164	0.183	0.266
NMF $> \Delta$	0.127	0.160	0.258	0.0477	0.152	0.244
$\text{RFDCM}_{\text{Basic}} > \Delta$	0.110	0.142	0.208	0.072	0.121	0.200
$\text{RFDCM}_{\text{LR}} > \Delta$	**0.201**	**0.259**	**0.349**	$\mathbf{0.164}^{\gamma}$	$\mathbf{0.201}^{\theta}$	**0.311**
% ***better than SVD***	+46.71	+15.11	+13.31	+56.19	-7.79	+3.66

Table 20: **MovieLens:** Ranking performance of the $\mathbf{RFDCM_{LR}}$ and the baseline methods using threshold values δ and Δ : The bold font denotes the best result in respective evaluation metric. β, θ, ζ , α, γ and δ indicate statistically significant improvements over $\mathbf{KNN^{\beta}_{ZScore+User}}$, $\mathbf{Co\text{-}Clustering}^{\theta}$, $\mathbf{SVD}^{\zeta}$, $\mathbf{SlopOne}^{\alpha}$ and $\mathbf{NMF}^{\gamma}$. The statistical significance is based on the paired t-test with p-value < 0.05.

Model	Books Crossing ($\delta = 3.0$, $\Delta = 5.0$)			LastFM ($\delta = 50.0$, $\Delta = 200.0$)		
	MAP@5	**MAP@10**	**HR@10**	**MAP@5**	**MAP@10**	**HR@10**
KNN$_{\text{Basic+[Item-based]}} > \delta$	0.038	0.076	0.164	0.246	0.438	0.753
KNN$_{\text{Basic+[User-based]}} > \delta$	0.037	0.079	0.162	0.229	0.418	0.748
KNN$_{\text{Means+[Item-based]}} > \delta$	0.031	0.830	0.165	0.299	0.465	0.755
KNN$_{\text{Means+[User-based]}} > \delta$	0.029	0.080	0.162	0.239	0.435	0.755
KNN$_{\text{ZScore+[Item-based]}} > \delta$	0.017	0.065	0.140	0.260	0.430	0.755
KNN$_{\text{ZScore+[User-based]}} > \delta$	0.016	0.063	0.137	0.246	0.438	0.753
Co-Clustering $> \delta$	0.053	0.110	0.175	0.282	0.463	0.775
SVD $> \delta$	0.0791	0.125	0.187	0.231	0.424	0.766
SlopeOne $> \delta$	0.051	0.072	0.141	0.303	0.468	0.767
NMF $> \delta$	0.042	0.056	0.112	0.268	0.421	0.750
RFDCM$_{Basic} > \delta$	0.024	0.077	0.150	0.366	0.507	0.771
RFDCM$_{LR} > \delta$	0.053^{β}	$\mathbf{0.189}^{\gamma}$	$\mathbf{0.337}^{\zeta\gamma}$	$\mathbf{0.420}^{\zeta\theta\beta\alpha\gamma}$	$\mathbf{0.553}^{\zeta\theta\beta\alpha\gamma}$	$\mathbf{0.794}^{\beta\gamma}$
% ***better than SVD***	-39.39	+40.76	+57.25	+58.44	+30.42	+3.65
KNN$_{\text{Basic+[Item-based]}} > \Delta$	0.038	0.084	0.160	0.295	0.456	0.746
KNN$_{\text{Basic+[User-based]}} > \Delta$	0.037	0.079	0.162	0.218	0.405	0.742
KNN$_{\text{Means+[Item-based]}} > \Delta$	0.035	0.080	0.167	0.287	0.459	0.753
KNN$_{\text{Means+[User-based]}} > \Delta$	0.029	0.080	0.162	0.261	0.454	0.760
KNN$_{\text{ZScore+[Item-based]}} > \Delta$	0.018	0.068	0.133	0.269	0.448	0.757
KNN$_{\text{ZScore+[User-based]}} > \Delta$	0.016	0.063	0.137	0.295	0.456	0.746
Co-Clustering $> \Delta$	0.048	0.103	0.150	0.247	0.449	0.771
SVD $> \Delta$	0.068	0.112	0.200	0.221	0.404	0.764
SlopeOne $> \Delta$	0.096	0.101	0.172	0.286	0.456	0.764
NMF $> \Delta$	0.095	0.057	0.137	0.254	0.409	0.75
RFDCM$_{\text{Basic}} > \Delta$	0.024	0.067	0.112	0.368	0.507	0.760
RFDCM$_{\text{LR}} > \Delta$	0.128	$\mathbf{0.138}^{\theta}$	$\mathbf{0.300}^{\zeta\theta}$	$\mathbf{0.389}^{\zeta\theta\beta\alpha\gamma}$	$\mathbf{0.545}^{\zeta\theta\beta\alpha\gamma}$	$\mathbf{0.803}^{\beta}$
% ***better than SVD***	+61.22	+20.80	+40.00	+76.01	+34.90	+5.10

Table 21: **Book Crossing and LastFM:** Performance of the **RFDCM$_{\text{LR}}$** and the baseline methods using threshold values δ and Δ : The bold font denotes the best result in respective evaluation metric. β, θ, ζ , α, γ and δ indicate statistically significant improvements over **KNN$^{\beta}_{\text{ZScore+User}}$**, **Co-Clustering**$^{\theta}$, **SVD**$^{\zeta}$, **SlopOne**$^{\alpha}$ and **NMF**$^{\gamma}$.

datasets. Interestingly, it worked impressively well with LastFM especially compared to MovieLens. The reason is that in contrast with MovieLens ratings, play counts are not restricted to a small range of numbers. In other words, term frequency is not limited, and this randomness fits better with IR and as burstiness. We also conducted the paired t-test with $p < 0.05$ to test the significance of improvements. Concerning MovieLens datasets, RFDCM_{LR} achieved it's best results on MovieLens-2018 using MAP@5. The results on LastFM show statistically significant improvements for all MAP measures. The differences of Average Precision shows that RFDCM_{LR} worked better for all individual queries except for a few instances. We analysed minimum, maximum and median MAPs of the model, SlopeOne, KNN, Co-Clustering, NMF and SVD. The analysis confirm the good performance of RFDCM_{LR} on LastFM.

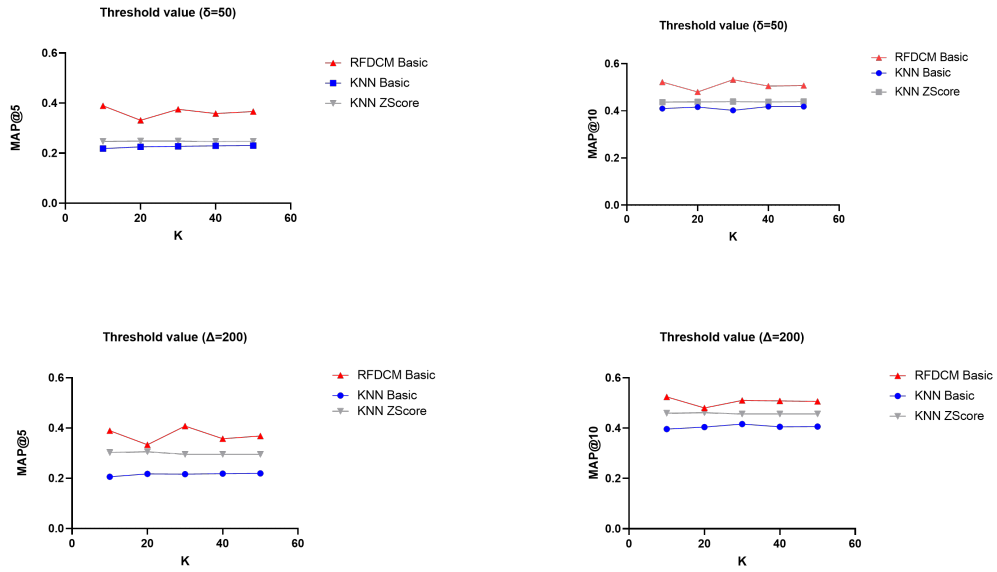
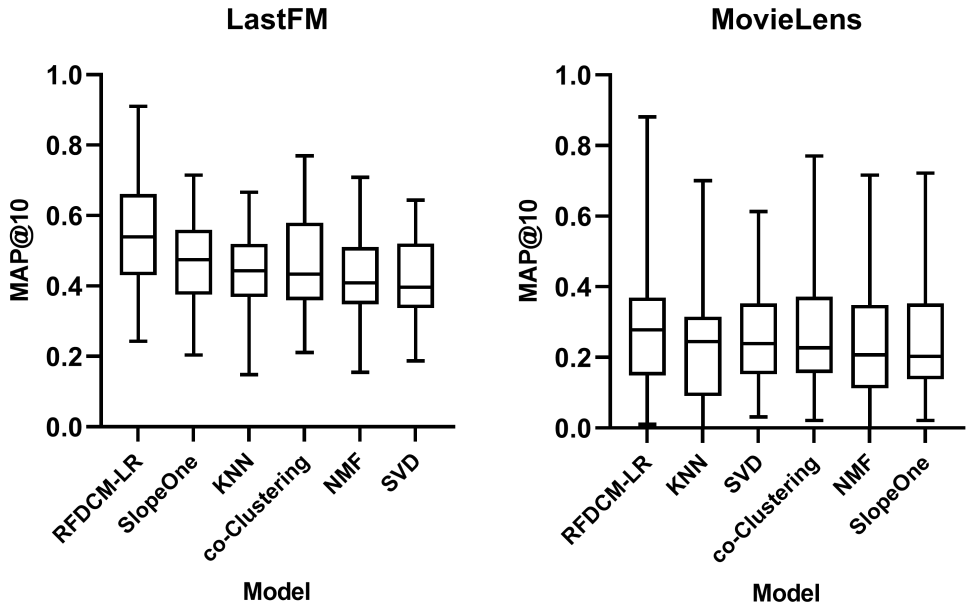


Figure 15: MAP@5 and MAP@10 of the KNN-based algorithms with different number of nearest neighbours (K) on LastFM.

Performance of KNN-based RFDCM

RFDCM_{LR} outweighed $\text{RFDCM}_{\text{Basic}}$ in all measures for both datasets. Consequently, while IR-based KNN is effective, the combination of IR features

Figure 16: Quality comparison of $\mathbf{RFDCM_{LR}}$ and baselines on LastFM and MovieLens.



with learning approaches leads to better results. Although $\text{RFDCM}_{\text{Basic}}$ was not able to outperform KNN baselines on MovieLens, it achieved a very good performance when applied to LastFM. This could be a benefit delivered by the use of play count as term frequency. Figure 15 compares the performance of $\text{RFDCM}_{\text{Basic}}$ with two other KNN baselines ($\text{KNN}_{\text{Basic}}$ and $\text{KNN}_{\text{ZScore}}$ against different number of neighbours (K). As can be seen, in contrast with KNN baselines, there is a fluctuation in the quality of RFDCM and changing the number of nearest neighbours affects the performance. Moreover, it turned out that the RFDCM variant works better when $k = 10$ and $k = 30$. The improved performance of $\text{RFDCM}_{\text{Basic}}$ was due to the simplicity of voting and measures concerning traditional KNN. Figure 16 confirms the good performance of RFDCM_{LR} on LastFM by analysing minimum, maximum and median MAPs of the model, SlopeOne, KNN, Co-Clustering, NMF and SVD.

6.2.4 Discussion

Traditional IR has been shown to be effective in collaborative filtering. However, one problem is that many approaches either rely on basic similarity measures for efficiency and feasibility reasons, or employ complex models that are difficult to explain. Moreover, it is difficult to represent the semantic aspects of the applications. Therefore, it is important to explore the applicability of advanced methods in IR to the recommendation task. The Dirichlet Compound Model (DCM) brings together ML and IR (both are important techniques in collaborative filtering). In this section, we developed an extension of DCM for recommendation (RFDCM) where user interactions are consolidated into semantic retrieval.

This section moves from semantic retrieval (FDCM in section 4.4) to recommendation. Subsequently, we introduced two instances of Recommender Full DCM (RFDCM) namely, $\text{RFDCM}_{\text{Basic}}$, eq (152), and RFDCM_{LR}, eq (153). URFDCM is an extension of FDCM (terms are items and users are documents) and developed to identify similar users. URFDCM is consumed by $\text{RFDCM}_{\text{Basic}}$ and RFDCM_{LR} to recommend items to users. The novelty in the development of the RFDCM is proposing effective IR-inspired features extracted from URFDCM results for training RFDCM_{LR}. The features for logistic regression are introduced in section 6.2.2. RFDCM converts user-interactions into documents and then effectively consolidates individual semantic scores of term-only, concept-only and user-interaction-only DCM methods. The more basic instance of RFDCM ($\text{RFDCM}_{\text{Basic}}$) explains the use of KNN similarity combined with URFDCM scores in recommendation (section 6.2.2).

RFDCM (recommender FDCM, section 6.2.1) leans on the FDCM approach. The main idea is to transfer concepts of IR (term-frequency) to the *recommendation task*. The RFDCM score builds upon the user-based RFDCM score (URFDCM, section 6.2.2. Similar to FDCM, URFDCM combines two scores: the item-based DCM (IDCM) score (section 6.2.2) and the concept-based DCM (CDCM) score from the FDCM model. The linear aggregation parameter σ_I (section 6.2.2) is the sibling of σ_T (the aggregation parameter for FDCM) in section 4.4.2. σ_I is derived from the semantic difference of max-idf values (whereas σ_T for semantic IR was based on sum-idf).

This section suggests that the use of play counts as term frequencies is beneficial and this representation of user interaction fits better with IR than the usage of ratings. Because term frequencies are high, methods like DCM

and other term frequency quantifications are conducive to recommendation systems.
Our experimental results showed that RFDCM has a positive effect on the determination of user neighbourhood. We trained a Logistic Regression method with IR features derived from URFDM results. The proposed model consistently outperformed collaborative filtering baselines (table 20 and table 21). Moreover, the learning approach outweighed IR-based KNN which confirms that the combination of IR and ML is effective.
In conclusion, this section highlights that IR and recommendation are closely related and adding to the evidence confirming that IR and recommendation are two sides of the same coin. Additionally, it highlights HOW advanced IR methods (Dirichlet model and semantic IR) can be effectively applied for recommendation.

Future work

Future work aims to develop a hybrid DCM-based filtering framework using the combination of proposed standards in this thesis including RFDCM, conceptual FDCM and opinion-aware models.
This framework transforms each user to vectors of corresponding weights of the semantic pillars including terms, opinions, concepts and user interactions. In other words, it creates individual formulated queries and representations of documents for any semantic pillar in the corpus.
The representation is generated from two heterogeneous documents per user so that the user has one textual document (e.g. profile and narratives) which is transformed to bag of words (BOW), bag of concepts (BOC) and bag of opinions (BOO) and one item-based document consists of items along with user interactions (e.g favourite articles and ratings or clicks). The item-based document satisfies the requirement for the creation of BOI (bag of items). Lets SDCM be the notation for the final retrieval score, ODCM, TDCM, CDCM and IDCM be notations of opinion-based, term-based, concept-based and item-based DCM, β be a sentiment bearing lexical feature and φ be a concept. The algorithm for this task is as follows:

$$\begin{aligned}&\mathrm{RSV}_{\mathrm{SDCM}}\left(d,q,c\right):=\\&\quad\sigma_t\cdot\mathrm{RSV}_{\mathrm{TDCM}}\left(d_t,q_t,c\right)+\sigma_o\cdot\mathrm{RSV}_{\mathrm{ODCM}}\left(d_\beta,q_\beta,c\right)\\&\quad+\sigma_c\cdot\mathrm{RSV}_{\mathrm{CDCM}}\left(d_\varphi,q_\varphi,c\right)+\sigma_i\cdot\mathrm{RSV}_{\mathrm{IDCM}}\left(d_\varphi,q_\varphi,c\right),\end{aligned}\tag{155}$$

d_x and q_x are formulated document and query representations for semantic pillar x (t, β and φ). σ variants denote aggregation parameters corresponding to the semantic types: σ_t is the parameter for the term, σ_o is the aggregation parameter for the opinion, σ_c is for concepts and σ_i is for items. A naive approach to estimate aggregation parameter is as follows (e.g. terms):

$$\sigma_t(q, \varphi, \beta) := \frac{n(t, q)}{n(t, q) + n(\varphi, q) + n(\beta, q)} \tag{156}$$

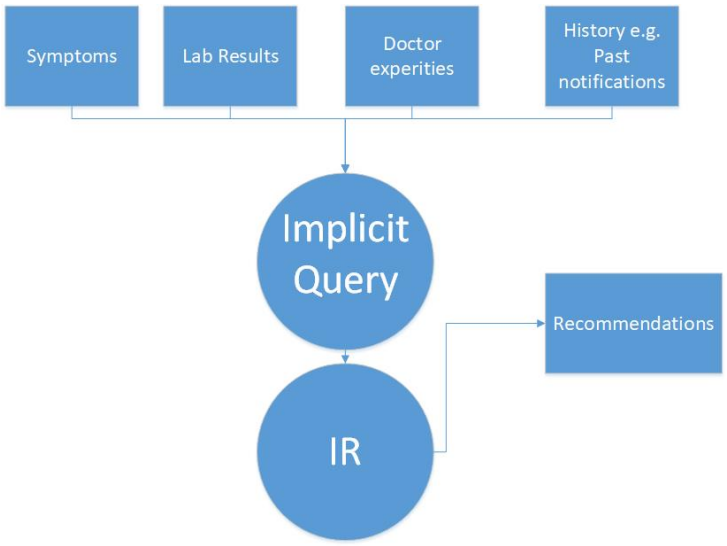
The recommender system then accordingly needs to be applied to a notification dataset. In addition to the urgent notification scenario, two example benchmarks suitable for the task are listed below:

- **Finding scientific papers:** Data sets such as Medline or OHSUMED could be considered to recommend citations to the researchers based on user preferences e.g. History of downloaded papers, the field of their research and favourite authors.

  Similar work has been done by [38]. They developed a biomedical recommendation system based on the author's collaboration network, productivity and area of research.
- **Recommendation of biomedical citations to doctors**: Doctors are always looking for relevant medical articles to be aware of the latest information in their field. Moreover, they need to be reading biomedical articles to treat their patients more effectively. In this case, the recommendation system could be personalized based on doctor expertise, patient's symptoms, lab tests and possible diagnosed disease.

We need to deal with the implicit query in which information needs from different sources are embedded. Figure 17 demonstrates an example of how a query generator can work for this task:

Figure 17: A recommendation system based on IR: Features are consolidated into an implicit query to be consumed by the IR framework.



6.3 ADOR: Sentiment-based medical dataset for IR

6.3.1 Background

Despite the fact that both sentiment analysis and IR are of importance with regards to medical applications, the work on incorporating sentiments into medical IR is limited, and there is no well-known benchmark established for this task. Many review-based datasets have been released for the task of sentiment analysis such as multi-domain Amazon dataset [62], INEX social book search [39] and IMDB dataset of reviews [69].
Several benchmarks have been published to examine reliability of different IR models with respect to medical applications including OHSUMED [45], CLEF-eHealth [39, 99]. Developing a sentiment-focused query set for a dataset such as OHSUMED is not optimal since documents are generated from medical literature. Although sentiments, e.g., *cancer* and *treatment* are included in documents, implications of urgency and feelings e.g., emojis are rarely found. Table 22 shows the overview of some well-known medical

datasets and lists fundamental statistics of their semantic features.
Sentiment analysis and opinion mining are popular research fields in natural language processing, data science and text mining. They analyse textual contents based on people's opinions, emotions and attitudes [67]. We introduce models which are semantic instances of a generalizable TF-IDF. The technology of semantic retrieval is of particular importance in medical applications and the integration of semantics with the standard content-based retrieval tools could lead to more intelligent search experiences [106, 8]. The generalization of TF-IDF towards semantic frameworks is discussed in [6]. When compared to retrieval systems built upon only bag of words, the integrated methods result in more performant question answering (QA) systems with constraint checking abilities. There has been research on developing conceptual models for medical applications [74] and [109]. It could be interesting to leverage sentiments and feelings in these applications. It could also be desirable to study the applicability of sentiment-based IR with respect to COVID datasets which I leave to future work.

Dataset	Reports	no.Queries	*avg*-opinions	*avg*-concepts
clef2013 e-health	ShARe CLEF eHealth Evaluation Lab 2013 [99]	50	0.3	2.9
clef2014 e-health	share-clef ehealth evaluation lab 2014 [55]	50	0.34	1.86
OHSUMED	OHSUMED [45] - TREC-9 Final Report [87]	63	0.41	4.87
TREC 2006 Genomics Track	TREC 2006 genomics track overview. [46]	27	0.32	6.00
TREC 2007 Genomics Track	TREC 2007 genomics track overview.	35	0.27	4.6

Table 22: Overview of well-established benchmarks for health-related retrieval.

6.3.2 Contribution

Features of dataset

The Amazon Dataset of Reviews (ADOR) is based on reviews from biomedical Amazon products derived from three super categories which are Medication & Remedies, Diagnostic and Monitoring Tools and Health-Related Books. We have defined a set of sub-category products inherited from the

super-categories and subsequently extracted reviews of related top ten items retrieved by Amazon search engine. However, in order to achieve a more balanced dataset in terms of polarity, we ignored items without negative reviews.

#Concepts	595442
#Distinct.Concepts	404748
#Opinions	194790
#Distinct.Opinions	163045
#Query	25
#Docs	44796
#Avg.Query Length	9.08
#Avg.Review.Text Length	35.38
#Sampling Date	31-03-2020

Table 23: The statistics of ADOR.

To make the data easily reusable, we followed two steps. Firstly, we converted the encoding of the contents to UTF-8 and secondly, we defined the schema and the required fields. The essential fields consists of Amazon ASIN number, medical category, star-rating, the title of the review, review text and labels including star-rating and helpful, have been embedded into the dataset.

ADOR Query Set

I have defined 25 topics based on five purposes. Figure 20 shows the distribution of queries and number of relevant documents. The five categories of information need are as follows:

1. The retrieval of positive or negative reviews associated with medical products.

2. Fact-based and non-sentiment-bearing queries which only intend to retrieve medical entities.

3. Ranking the polarity of item-reviews within the sub-categories, e.g. vitiligo cream and flu tablets.

4. Ranking the polarity of item-reviews within the super-categories, e.g. medications or diagnostic tools.

5. The retrieval of extreme (most positive or most negative) reviews given different medical concepts. We used modifiers to give attention to the information need, e.g. *Highly negative reviews for books about borderline personality disorder.*

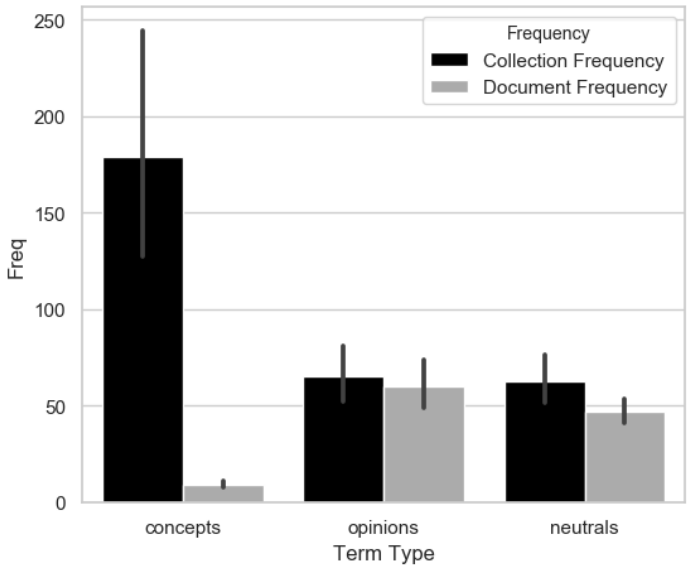


Figure 18: Document and collection statistics of the ADOR semantic types: The opinions group has the highest document frequency.

Overview of ADOR

In this section, we briefly present the dataset and provide the statistics of ADOR. Table 23 lists the fundamental statistics of the dataset. There are 194790 opinions or sentiment-bearing words (e.g fatigue, great and very bad) and 59442 medical concepts (e.g flu) in the dataset which is distributed across 44796 documents. We used VADER lexicon to capture opinions and MetaMap to bind terms to medical concepts. Figure 19 presents the distribution of document length and query length. The majority of queries (more than 35%) have a length between 9 and 12 words. More than 50% of documents have between 1 and 20 words, whereas 7% of them are longer than 100 words. The statistics regarding distribution of queries and their relevant documents are shown in figure 20. As can be seen, 28% of queries contain 1-60 relevant documents which is the exact same percentage for queries with more than 240 relevant documents. The rest of the queries contain between 60 and 240 relevant documents. We extracted the average document and collection frequencies of semantic types (neutral terms, concepts and opinions) of the ADOR which can be found in figure 18. Even though the average document frequency of opinions is high, opinions could significantly impact the retrieval quality due to the nature of reviews.

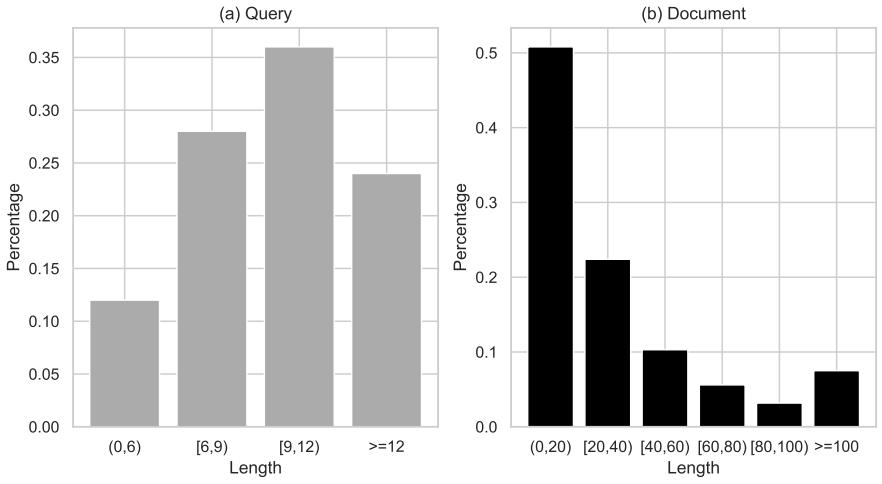


Figure 19: The distribution of document length and query length.

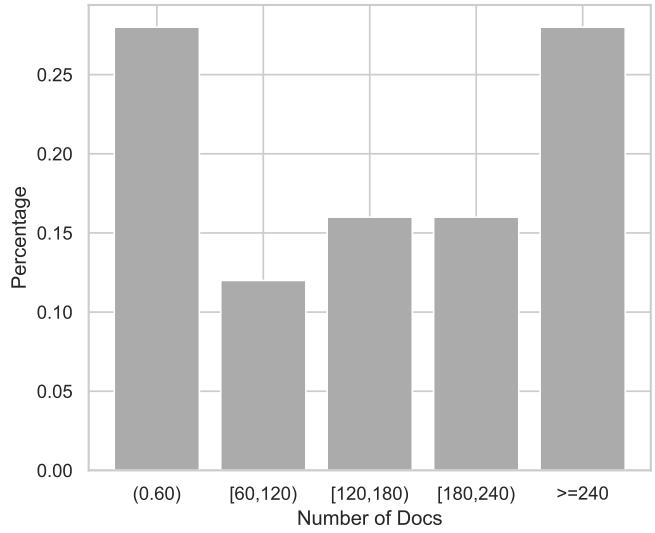


Figure 20: The distribution of queries and number of relevant documents.

Application of the benchmark

Rationales Although the use of human judgments could seem ideal for the generation of gold standards, we developed a generic framework which has some privileges, e.g., it could be easily used to build gold standards for new query sets.

We provided informative labels, including rating-star, the number of people who found reviews helpful and medical categories of Amazon products when preparing the data. This framework helps to rapidly develop new queries that could be formulated into the provided labels. Considering the example query *Why do some customers are happy with books about caffeine addiction and narcissistic personality disorder.*, the formulated query is : (Rating=[4,5], Super-Category=[Books], Sub-Category=[NPD,Caffeine Addiction]). In other words, any review in the dataset that meets the information needs requested by the formulated query could be selected.

To evaluate the accuracy of models, one approach would be the use of existing reviews as queries. However, there are two substantial issues with this approach. Firstly, data scientists need to analyse and classify their experimental results based on the query intent, e.g. fact-based, binary and

explorative queries. The use of reviews as queries is not in line with the nature of query intent. Secondly, reviews are strongly focused on opinions. Therefore, generating a robust query set consists of a balanced combination of concepts, terms and opinions do interfere with the structure of reviews.

Baseline Models Ranking algorithms are the primary baselines. However, the benchmark is also able to be used for the prediction/classification tasks. For example, a review could be considered as a message posted by a patient or a customer. In this case, the evaluation approach is to predict if it is extreme (very negative) and requires attention by an expert, e.g., doctor, nurse or a company member. The other applicable task is notification systems. In this scenario, users post messages and an algorithm needs to decide who (e.g. which doctor, expert) should be notified for analysing the message or responding to it.
Furthermore, the framework could be employed by data scientists to predict features provided by the dataset such as positive/negative and helpful/not helpful. Baselines could be used such as Neural Network classifier (e.g., Bert or scikit), Bayesian predictor, regression and K-NN (nearest neighbours) to measure the q prediction quality. The KNN classifier could be applied to retrieve the most similar train reviews (e.g., cosine similarity), aggregate evidence and assigns a label to the test review.

Processing the New Queries To confirm the capability of the benchmark with models derived from opinions and concepts, we have developed a semantic retrieval baseline for this section. We briefly describe the methodology and then show the experimental results for comparing the semantic approach with traditional and recent IR methods on ADOR.

6.3.3 Evaluation

Methodology

Our approach is to leverage the well-known TF-IDF and capture its semantic extensions which are built upon opinions and/or concepts. To make the formulations readable, we use type-aware x functions, e.g. $\mathrm{OF}(o, d)$ is the opinion frequency of opinion o in document d, where $\mathrm{CF}(\varphi, d)$ is the frequency of concept φ in the document. Let q be a query, d be a document

Model	**Evaluation Measure**			
	P@5	**P@10**	**NDCG**	**MAP**
TF-IDF	0.2480	0.2720	0.2354	0.0833
BM25	0.3120	0.3160	0.2336	0.0813
KNRM	0.2320	0.2440	0.2445	0.0906
DSSM	0.2080	0.2200	0.2422	0.1039
arc-I	0.3520	0.3040	0.2476	0.0902
CF.IDF	0.3840	0.4080	0.2619	0.1106
OF.IDF	0.3680	0.4120	0.2758	0.1250
OF.IDF+TF-IDF (*w=0.5*)	0.3600	0.3920	0.2705	0.1175
OF.IDF+CF.IDF (*w=0.5*)	**0.4640**$^{\beta\theta\zeta}$	**0.4280**$^{\beta\theta\zeta}$	**0.2825**$^{\beta\theta\zeta}$	**0.1274**$^{\beta\theta}$

Table 24: Ranking performances of the opinion-aware models and the baseline methods: The bold font denotes the best result in that evaluation metric. β, θ, ζ indicate statistically significant improvements of the best model over **BM25**$^{\beta}$, **KNRM**$^{\theta}$ and **DSSM**$^{\zeta}$. The statistically significance is based on the paired t-test with p-value < 0.05.

and let c be the collection, the Retrieval Status Value (RSV) of the opinion-aware model is as follows:

$$\text{RSV}_{\text{OF-IDF}}(d, q, c) := \sum_{o \in t} \text{OF}(o, q) \cdot \text{OF}(o, d) \cdot \text{IDF}(o, c) \tag{157}$$

IDF (o, c) is the Inverse Document Frequency of the opinion o in the collection. t is a list of all lexical features in lexicon where the sentiment polarity is equal to query polarity. For example, given query *Any useless or poor medications for allergy or cold sore.*, the query polarity is negative, and consequently, the t list comprises all negative opinions in the lexicon.
Let φ be a medical concept and let IDF (φ, c) be the Inverse Document Frequency weight of the concept, the conceptual extension of TF-IDF is defined as below:

$$\text{RSV}_{\text{CF-IDF}}(d, q, c) := \sum_{\varphi \in q} \text{CF}(\varphi, q) \cdot \text{CF}(\varphi, d) \cdot \text{IDF}(\varphi, c) \tag{158}$$

Analysis of results

We briefly analyse the evaluation results of ADOR against the semantic approach introduced in this section, TF-IDF, BM25 and some recent neural ranking models. We have trained neural ranking models including KNRM [115], DSSM [50] and arc-I [48] on ADOR. We performed 5-fold cross-validation where the final fold in each run was considered as the test set. We randomly divided queries into five-folds and repeatedly captured the average of the fivefold-level evaluation results. All neural models were developed using MatchZoo [30] based on *tensorflow* with Adam optimizer, batch size 16 and learning rate=0.001. Using the Lucene framework and the language modelling with Dirichlet Prior, we retrieved pseudo-relevant documents and subsequently, the top 100 documents were re-ranked by the models. In addition to OF.IDF and CF.IDF, we conducted experiments on linear combinations of opinion-aware TF-IDF with term-based and conceptual TF-IDF using aggregation parameter $w = 0.5$. Concerning concept-based models, we

used MetaMap to extract concepts accompanied by their frequencies, semantic types and scores. We counted 'trigger' attributes of MetaMap-outputs to calculate the corresponding frequencies of semantic types.
Table 24 shows the experimental results on ADOR using four metrics including P@5, p@10, NDCG and Mean Average Precision (MAP). We also conducted the paired t-test with $p < 0.05$ to compute the significance of improvements. The isolated OF.IDF and CF.IDF worked better than TF-IDF, BM25 and neural models (KNRM, DSSM, arc-I) while the combination of opinions and concepts received the best results. The interesting finding is that the models based on combinations of opinions with both terms (OF.IDF+TF-IDF) and concepts (OF.IDF+CF.IDF) improved all the measures.

6.3.4 Discussion

Sentiment analysis has received attention in retrieval applications. Combining opinions such as user feelings with semantics would enhance the performance of these applications, especially when the level of urgency is essential, e.g., medical domain. However, no widely medical benchmark is known for evaluating sentiment-aware IR.
In this section, we introduced a new benchmark, namely ADOR which is a subset of Amazon reviews for medical products and made it publicly available. For our research aim, the dataset allows for bringing and testing sentiment-based IR to medical domain. The corresponding dataset focuses on medical products within three categories including medicine, monitoring tools and health-related books. The collection of reviews comes with a structured framework which enables users to automatically generate relevance labels for new topics. Moreover, a query set with relevance results was consolidated into the benchmark. In order to develop this query set, we considered factors such as query intent, sentiment score of query and concept query frequency.
To assess the compatibility of the benchmark with semantics (opinions and concepts) and also measure the suitability of the benchmark for sentiment-based IR we proposed the sentiment-aware extension of TF-IDF and applied it to the dataset. The models are naive but reproducible and retrieve results using linear combinations of sentiment-only TF-IDF score, term-only TF-IDF score and concept-only TF-IDF score.
We compared the new approach with well-established and modern retrieval

models. Our experiments confirmed that the integration of sentiments with IR improves the quality of ranking with regards to the ADOR dataset. The semantic model based on combination of OF.IDF and CF.IDF achieved the best results against gold standards.
In conclusion, the ADOR benchmark could help researchers to develop and evaluate opinion-aware retrieval models. This benchmark could help health-care organizations and companies to effectively detect, rank and filter urgent notifications based on patient's health status, narratives and conditions. The benchmark is available at https://github.com/mb320/ADOR.

Chapter 7

Summary

7.1 Research contribution

1. **Semantic-based DCM (FDCM)** In section 4.4, we proposed a DCM-based retrieval framework (FDCM) in order to investigate hypothesis 2 and technical aims 1 and 2. FDCM is an extension of DCM which is built upon the aggregation of term-only and concept-only DCM scores. The development of the aggregation parameter and respective candidates are shown in definition (9)(page 87), and the FDCM ranking formula is presented in definition (8)(page 81). FDCM is a conceptual probabilistic standard which is a crucial dimension in improving urgent notification filtering.

2. **Recommender-based DCM (RFDCM)** In section 6.2, we developed an extension of the FDCM framework, namely RFDCM, specifically for the recommendation task. RFDCM was proposed to satisfy hypothesis 1 and technical aims 5, 6 and 7. RFDCM hires user interactions such as ratings and play counts in order to bring IR to recommendation (notification filtering) task. This standard is capable of easily being combined with other semantic instances of FDCM. The user-based RFDCM (URFDCM) is presented in definition (14)(page 115) and corresponding top-N recommendation models for similar items are presented in eq (152) and eq (153)(page 116).

3. **IR models based on opinion-oriented TF ($\mathbf{TF_{intense}}$, $\mathbf{TF_{sentiment}}$)** We proposed a set of novel opinion-aware IR mod-

els in section 5.2. These semantic instances of IR (LM and TF-IDF variants) were proposed to investigate hypothesis 2 and technical aim 3. We proposed a mapping to transfer opinions to terms in classic IR, and the sentiment intensity of the opinion to term frequency. We showed the opinion-aware term frequency variants in definition (10)(page 98) and definition (12)(page 99). This approach builds the grounds for using opinion-aware IR for consideration of the urgency element in notification filtering.

4. **ADOR medical benchmark** In section 6.3, ADOR is created as a new medical benchmark to investigate hypothesis 2 and technical aim 4. ADOR is generated based on Amazon reviews of medical products. The importance of such a benchmark is the feasibility of having terms, opinions and concepts altogether in collection and queries. ADOR enables data scientists to evaluate the quality of aggregated semantic IR models (terms+concepts+opinions).

7.2 Limitations

In our research we needed a dataset that accommodates sentiment-based, concept-based and term-based retrieval and additionally be suitable for the evaluation of recommender systems. Unfortunately, we were not able to find such a dataset which is publicly available and therefore, we utilized specific datasets for each task.
My hypothesis is to develop a fully IR-based recommender system (notification filtering tool) which consists of two components: document ranking (text-based e.g. user profile or narratives) and collaborative filtering (e.g recommending items to GP and patients based on user interaction and similarity). Confirming the effectiveness of each component individually infers that combining them results in a reliable solution for the research question. To evaluate the hypothesis, I used designated datasets for each component as detailed below:

Document Ranking

- **Conceptual standards:** We used well-established benchmarks for biomedical data including TREC and OHSUMED to evaluate both concept-only and aggregated (concepts and terms) approaches.

- **Opinion-aware standards:** We confirmed the effectiveness of opinion-based IR against datasets such as IMDB and Amazon reviews of movies. Review datasets are rich in sentiments and thus can be considered as suitable candidates for the evaluation of opinion-aware retrieval.

- **Semantically aggregated standards:** We needed a benchmark for analysing models derived from combining terms, concepts and opinions. Therefore, we published a benchmark based on Amazon reviews of medical products to confirm the capability of aggregated methods with textual documents containing terms, concepts and opinions.

Collaborative Filtering

- **IR-based Recommender:** We evaluate the recommender component against well-known benchmarks including MovieLens, Book Crossing and LastFM.

7.3 Conclusion

We explored advanced IR as a solution to urgent notification filtering by using Dirichlet-multinomial distribution in the context of semantic IR and KNN-based recommendation. IR is typically applied for the sub-task of retrieving related items in respect to recommendation where a common similarity measure (sometimes simply a vector similarity) is applied instead of more advanced models. However, in this thesis, we emphasised the usage of advanced models, focusing on the Dirichlet-multinomial approach (DCM). The focus of my work was developing a generalizable, transparent and light-weighted framework which is not just term-based, but is the result of the aggregation of term-based, concept-based, opinion-based and user interaction-based (e.g. rating) aspects of IR. DCM was chosen since it is probabilistically well-defined, gives rise to burstiness and brings together ML and IR which are both substantial techniques in predication tasks. This thesis highlighted the importance of establishing generalizable standards for urgent notification (recommendation) filtering by bridging the gap between text-based ranking and recommendation.
Recommendation/notification are critical in many areas such as medical and criminal domains where a failure in detecting urgent notifications could take

a heavy toll on people. It is therefore important to apply transparent, analytical models where it is possible to induce WHY a document has been retrieved, WHY a recommendation will be made and WHY a notification will be sent. Only the usage of clear, widely established mathematical standards can guarantee such transparency, a transparency that is not only for *few experts*, but for the *many data scientists* applying retrieval and recommendation algorithms. While there are quasi-standards for term-only IR (TF-IDF, BM25, multinomial and Dirichlet-multinomial language modelling, and even particular variants of DFR (divergence from randomness)), there is no established standard for semantic IR. This hinders the usage of semantic IR approaches in disciplines that rely on IR, and we reached out in this thesis to recommendation and notification.

Our proposed framework firstly separates terms and concepts, and then integrates them into an overall Full DCM (FDCM) score. This is the result of effective transformation from term-based DCM (TDCM) over conceptual DCM (CDCM) to FDCM. Critical for the integration is the parameter for aggregating TDCM and CDCM. One of the substantial contributions of this thesis was the estimation of the aggregation parameter for FDCM. We compared a set of aggregation candidates that varied in effect regarding a query factor based on Query Performance Prediction (QPP). The experimental results showed that the best candidate was $\sigma_{T,\text{raw-sem},1}$, which employed a logarithmic adjustment of the impact of TDCM and CDCM. Moreover, FDCM consistently dominated the language modelling baselines and was shown to be more effective than both terms-only and concepts-only DCM models.

Subsequently, we introduced Recommender Full DCM (RFDCM) which converts documents into bag of user interactions and accordingly consolidates individual semantic scores of item-only, concept-only and user interaction-only ranking scores. RFDCM is a FDCM inspired approach in which concepts of IR are transferred to the recommendation task. Our experimental results showed that RFDCM has a positive effect on the determination of user neighbourhood (top similar users). We trained a Logistic Regression (LR) model with IR features derived from an analysis of User-based RFDCM (URFDCM) results. The LR approach consistently outperformed collaborative filtering baselines. Moreover, it outperformed IR-based KNN which suggests that the combination of IR and ML can enhance the quality of filtering.

Moreover, this thesis proposed opinion-aware retrieval as an effective tool in taking urgency in consideration with regards to notification filtering. In this thesis, urgency refers to the degree to which the notification needs imme-

diate attention. For example, concerning the medical domain, it could be interpreted as a measure to predict if a patient is in need of urgent assistance (should a notification immediately be sent out to the doctor or patient). The determination of urgency or criticality is not only about relevance (term-based approach) and needs more requirements. By answering the hidden sentiment analysis task in such domains, the urgency of a notification could be predicted more precisely than using only relevance-based IR (e.g term-only models). We presented two novel families of opinion-aware models, namely sentiment-aware and intensity-aware models to deal with the problem of the opinion words with low IDF and high intensity. We discussed the consideration of a notion of IDF in sentiment classifications. To investigate the use of sentiment intensity in retrieval, we applied both basic and intensity-aware models to movie reviews and confirmed the effectiveness of the proposed approach. Furthermore, we created a benchmark based on Amazon reviews of medical products with a balanced occurrence of concepts and opinions. The importance of establishing such benchmark as well as the so-called opinion-aware retrieval is to encourage leveraging analytical models for the detection of urgency. We hope the consideration of opinions as a semantic dimension in IR benefits future research in notification filtering and leads to positive results especially when the models are combined with other semantic methods (FDCM and RFDCM).
In conclusion, this thesis paved the way to establishing and leveraging IR standards for urgent notification filtering. By bringing together recommendation (prediction) and retrieval, we lay the grounds for generalizable and hybrid algorithms. Additionally, it highlighted that IR and recommendation are closely related, and that advanced IR methods (Dirichlet model and semantic IR) are beneficial for recommendation. Establishing links between IR, ML and AI is important for achieving algorithms that can be viewed as transparent, and can therefore contribute to the standardization of algorithms.

7.4 Future work

Future work focuses on improving semantic ranking standards and developing a practical and generalizable notification filtering tool for urgent domains.

- Reinforce traditional IDF to be opinion-aware by using intensity frequency and adjusted document frequency. Possible candidates for the

opinion-aware IDF are discussed in section 5.2.4.

- Develop and evaluate new aggregation parameters for FDCM retrieval scores (section 4.4.4).
- Aggregate the weights of semantic parings as shown in section 4.4.4 (instead of retrieval scores).
- Develop a final hybrid recommender system based on IR which takes into consideration all approaches proposed in the thesis (section 6.2.4).

Bibliography

[1] Abacha, A.B., Zweigenbaum, P.: Automatic extraction of semantic relations between medical entities: a rule based approach. Journal of biomedical semantics **2**(5), S4 (2011)

[2] Adomavicius, G., Tuzhilin, A.: Toward the next generation of recommender systems: A survey of the state-of-the-art and possible extensions. IEEE transactions on knowledge and data engineering **17**(6), 734–749 (2005)

[3] Agosti, M., Di Nunzio, G.M., Marchesin, S., Silvello, G.: Medical retrieval using structured information extracted from knowledge bases. Proc. of the 27th SEBD (in print) (2019)

[4] Aizawa, A.: An information-theoretic perspective of tf–idf measures. Information Processing & Management **39**(1), 45–65 (2003)

[5] Aronson, A.R.: The mmi ranking function. Available in the website: https://ii. nlm. nih. gov/MTI/Details/mmi. shtml (1997)

[6] Azzam, H., Yahyaei, S., Bonzanini, M., Roelleke, T.: A schema-driven approach for knowledge-oriented retrieval and query formulation. In: Proceedings of the Third International Workshop on Keyword Search on Structured Data. pp. 39–46. ACM (2012)

[7] Bahrani, M., Roelleke, T.: Novel query performance predictors and their correlations for medical applications. In: Proc. 1st International Workshop on Generalization in Information Retrieval (GLARE 2018). http://glare2018. dei. unipd. it/paper/glare2018-paper3. pdf (2018)

[8] Bahrani, M., Roelleke, T.: FDCM: Towards balanced and generalizable concept-based models for effective medical ranking. In: Proceedings of

the 29th ACM International Conference on Information & Knowledge Management. pp. 1957–1960 (2020)

[9] Banerjee, A., Dhillon, I., Ghosh, J., Merugu, S., Modha, D.S.: A generalized maximum entropy approach to bregman co-clustering and matrix approximation. The Journal of Machine Learning Research **8**, 1919–1986 (2007)

[10] Beineke, P., Hastie, T., Manning, C., Vaithyanathan, S.: Exploring sentiment summarization. In: Proceedings of the AAAI spring symposium on exploring attitude and affect in text: theories and applications. vol. 39 (2004)

[11] Belkin, N.J., Croft, W.B.: Information filtering and information retrieval: Two sides of the same coin? Communications of the ACM **35**, 29–38 (1992)

[12] Bellogín, A., Castells, P., Cantador, I.: Statistical biases in information retrieval metrics for recommender systems. Information Retrieval Journal **20**(6), 606–634 (2017)

[13] Bellogín, A., Wang, J., Castells, P.: Bridging memory-based collaborative filtering and text retrieval. Information Retrieval pp. 1–28 (2012)

[14] Bhatt, C., Dey, N., Ashour, A.S.: Internet of things and big data technologies for next generation healthcare. Springer (2017)

[15] Blitzer, J., Dredze, M., Pereira, F.: Biographies, bollywood, boomboxes and blenders: Domain adaptation for sentiment classification. In: Proceedings of the 45th annual meeting of the association of computational linguistics. pp. 440–447 (2007)

[16] Bonzanini, M., Martinez-Alvarez, M., Roelleke, T.: Opinion summarisation through sentence extraction: An investigation with movie reviews. In: Proceedings of the 35th international ACM SIGIR conference on Research and development in information retrieval. pp. 1121–1122. ACM (2012)

[17] Bonzanini, M., Martinez-Alvarez, M., Roelleke, T.: Extractive summarisation via sentence removal: Condensing relevant sentences into a short summary. In: Proceedings of the 36th international ACM SIGIR

conference on Research and development in information retrieval. pp. 893–896. ACM (2013)

[18] Cai, C., Wang, L.: Application of improved k-means k-nearest neighbor algorithm in the movie recommendation system. In: 2020 13th International Symposium on Computational Intelligence and Design (ISCID). pp. 314–317. IEEE (2020)

[19] Cantador, I., Brusilovsky, P., Kuflik, T.: 2nd workshop on information heterogeneity and fusion in recommender systems (hetrec 2011). In: Proceedings of the 5th ACM conference on Recommender systems. RecSys 2011, ACM, New York, NY, USA (2011)

[20] Carmel, D., Yom-Tov, E.: Estimating the query difficulty for information retrieval. Synthesis Lectures on Information Concepts, Retrieval, and Services **2**(1), 1–89 (2010)

[21] Castells, P., Fernandez, M., Vallet, D.: An adaptation of the vector-space model for ontology-based information retrieval. IEEE Transactions on Knowledge and Data Engineering **19**(2), 261–272 (2006)

[22] Cronen-Townsend, S., Zhou, Y., Croft, W.B.: Predicting query performance. In: Proceedings of the 25th annual international ACM SIGIR conference on Research and development in information retrieval. pp. 299–306. ACM (2002)

[23] Cummins, R., Paik, J.H., Lv, Y.: A pólya urn document language model for improved information retrieval. ACM Transactions on Information Systems (TOIS) **33**(4), 21 (2015)

[24] Davidov, D., Tsur, O., Rappoport, A.: Enhanced sentiment learning using twitter hashtags and smileys. In: Proceedings of the 23rd international conference on computational linguistics: posters. pp. 241–249. Association for Computational Linguistics (2010)

[25] Deerwester, S., Dumais, S., Furnas, G., Landauer, T., Harshman, R.: Indexing by latent semantic analysis. Journal of the American Society for Information Science **41**(6), 391–407 (1990)

[26] Ebesu, T., Shen, B., Fang, Y.: Collaborative memory network for recommendation systems. In: The 41st International ACM SIGIR Conference on Research & Development in Information Retrieval. pp. 515–524 (2018)

[27] Egozi, O., Markovitch, S., Gabrilovich, E.: Concept-based information retrieval using explicit semantic analysis. ACM Transactions on Information Systems (TOIS) **29**(2), 1–34 (2011)

[28] Elkan, C.: Clustering documents with an exponential-family approximation of the dirichlet compound multinomial distribution. In: Cohen, W.W., Moore, A.W. (eds.) Machine Learning, Proceedings of the Twenty-Third International Conference (ICML 2006), Pittsburgh, Pennsylvania, USA, June 25-29, 2006. ACM International Conference Proceeding Series, vol. 148, pp. 289–296. ACM (2006). https://doi.org/10.1145/1143844.1143881, https://doi.org/10.1145/1143844.1143881

[29] Esuli, A., Sebastiani, F.: Determining the semantic orientation of terms through gloss classification. In: Proceedings of the 14th ACM international conference on Information and knowledge management. pp. 617–624. ACM (2005)

[30] Fan, Y., Pang, L., Hou, J., Guo, J., Lan, Y., Cheng, X.: Matchzoo: A toolkit for deep text matching. arXiv preprint arXiv:1707.07270 (2017)

[31] Fang, H., Zhai, C.: An exploration of axiomatic approaches to information retrieval. In: Baeza-Yates, R.A., Ziviani, N., Marchionini, G., Moffat, A., Tait, J. (eds.) SIGIR 2005: Proceedings of the 28th Annual International ACM SIGIR Conference on Research and Development in Information Retrieval, Salvador, Brazil, August 15-19, 2005. pp. 480–487. ACM (2005). https://doi.org/10.1145/1076034.1076116, https://doi.org/10.1145/1076034.1076116

[32] Frommholz, I., Fuhr, N.: Probabilistic, object-oriented logics for annotation-based retrieval in digital libraries. In: Proceedings of the 6th ACM/IEEE-CS joint conference on Digital libraries. pp. 55–64 (2006)

[33] Gao, Y., Xu, Y., Li, Y.: Pattern-based topics for document modelling in information filtering. IEEE Transactions on Knowledge and Data Engineering **27**(6), 1629–1642 (2014)

[34] George, T., Merugu, S.: A scalable collaborative filtering framework based on co-clustering. In: Fifth IEEE International Conference on Data Mining (ICDM'05). pp. 4–pp. IEEE (2005)

[35] Gerani, S., Carman, M.J., Crestani, F.: Proximity-based opinion retrieval. In: Proceedings of the conference on research and development in information retrieval. ACM (2010)

[36] Go, A., Bhayani, R., Huang, L.: Twitter sentiment classification using distant supervision. CS224N Project Report, Stanford **1**(12), 2009 (2009)

[37] Gotlieb, C., Kumar, S.: Semantic clustering of index terms. Journal of the ACM **15**(4), 493–513 (Oct 1968)

[38] Guerra, J., Quan, W., Li, K., Ahumada, L., Winston, F., Desai, B.: Scosy: A biomedical collaboration recommendation system. In: 2018 40th Annual International Conference of the IEEE Engineering in Medicine and Biology Society (EMBC). pp. 3987–3990. IEEE (2018)

[39] Hall, M., Huurdemann, H., Skov, M., Walsh, D., et al.: Overview of the inex 2014 interactive social book search track. In: Conference & Labs of the Evaluation Forum (CLEF) (2014)

[40] Harper, F.M., Konstan, J.A.: The movielens datasets: History and context. ACM Trans. Interact. Intell. Syst. **5**(4), 19:1–19:19 (2016). https://doi.org/10.1145/2827872, https://doi.org/10.1145/2827872

[41] Hauff, C., Azzopardi, L., Hiemstra, D.: The Combination and Evaluation of Query Performance Prediction Methods. In: Proceedings of the 31st European Conference on Information Retrieval (ECIR). pp. 301–312. Springer-Verlag (2009)

[42] He, B., Macdonald, C., He, J., Ounis, I.: An effective statistical approach to blog post opinion retrieval. In: Proceedings of the 17th ACM

conference on Information and knowledge management. pp. 1063–1072. ACM (2008)

[43] He, B., Ounis, I.: Query performance prediction. Information Systems **31**(7), 585–594 (2006)

[44] Herlocker, J., Konstan, J.A., Riedl, J.: An empirical analysis of design choices in neighborhood-based collaborative filtering algorithms. Information retrieval **5**(4), 287–310 (2002)

[45] Hersh, W., Buckley, C., Leone, T., Hickam, D.: Ohsumed: an interactive retrieval evaluation and new large test collection for research. In: SIGIR'94. pp. 192–201. Springer (1994)

[46] Hersh, W.R., Cohen, A.M., Roberts, P.M., Rekapalli, H.K.: Trec 2006 genomics track overview. In: TREC. vol. 7, pp. 500–274 (2006)

[47] Hiemstra, D.: A probabilistic justification for using tf.idf term weighting in information retrieval. International Journal on Digital Libraries **3(2)**, 131–139 (2000)

[48] Hu, B., Lu, Z., Li, H., Chen, Q.: Convolutional neural network architectures for matching natural language sentences. In: Advances in neural information processing systems. pp. 2042–2050 (2014)

[49] Hu, B., Shi, C., Zhao, W.X., Yu, P.S.: Leveraging meta-path based context for top-n recommendation with a neural co-attention model. In: Proceedings of the 24th ACM SIGKDD International Conference on Knowledge Discovery & Data Mining. pp. 1531–1540 (2018)

[50] Huang, P.S., He, X., Gao, J., Deng, L., Acero, A., Heck, L.: Learning deep structured semantic models for web search using clickthrough data. In: Proceedings of the 22nd ACM international conference on Information & Knowledge Management. pp. 2333–2338 (2013)

[51] Huang, X., Croft, W.B.: A unified relevance model for opinion retrieval. In: Proceedings of the 18th ACM conference on Information and knowledge management. pp. 947–956. ACM (2009)

[52] Hug, N.: Surprise: A python library for recommender systems. Journal of Open Source Software **5**(52), 2174 (2020)

[53] Hutto, C.J., Gilbert, E.: Vader: A parsimonious rule-based model for sentiment analysis of social media text. In: Eighth international AAAI conference on weblogs and social media (2014)

[54] Kamps, J., Marx, M., Mokken, R.J., De Rijke, M., et al.: Using wordnet to measure semantic orientations of adjectives. In: LREC. vol. 4, pp. 1115–1118. Citeseer (2004)

[55] Kelly, L., Goeuriot, L., Suominen, H., Schreck, T., Leroy, G., Mowery, D.L., Velupillai, S., Chapman, W.W., Martinez, D., Zuccon, G., et al.: Overview of the share/clef ehealth evaluation lab 2014. In: International Conference of the Cross-Language Evaluation Forum for European Languages. pp. 172–191. Springer (2014)

[56] Kennedy, A., Inkpen, D.: Sentiment classification of movie reviews using contextual valence shifters. Computational intelligence **22**(2), 110–125 (2006)

[57] Koopman, B., Zuccon, G., Bruza, P.: What makes an effective clinical query and querier? Journal of the Association for Information Science and Technology **68**(11), 2557–2571 (2017)

[58] Kurland, O., Shtok, A., Carmel, D., Hummel, S.: A unified framework for post-retrieval query-performance prediction. In: Proceedings of the 3rd International Conference on Information Retrieval Theory (ICTIR). pp. 15–26. Springer-Verlag (2011)

[59] Lee, J., Yoon, W., Kim, S., Kim, D., Kim, S., So, C.H., Kang, J.: Biobert: pre-trained biomedical language representation model for biomedical text mining. arXiv preprint arXiv:1901.08746 (2019)

[60] Lemire, D., Maclachlan, A.: Slope one predictors for online rating-based collaborative filtering. In: Proceedings of the 2005 SIAM International Conference on Data Mining. pp. 471–475. SIAM (2005)

[61] Li, B., Wan, S., Xia, H., Qian, F.: The research for recommendation system based on improved knn algorithm. In: 2020 IEEE International Conference on Advances in Electrical Engineering and Computer Applications (AEECA). pp. 796–798. IEEE (2020)

[62] Li, S., Zong, C.: Multi-domain sentiment classification. In: Proceedings of ACL-08: HLT, Short Papers. pp. 257–260 (2008)

[63] Li, X., She, J.: Collaborative variational autoencoder for recommender systems. In: Proceedings of the 23rd ACM SIGKDD international conference on knowledge discovery and data mining. pp. 305–314 (2017)

[64] Limsopatham, N., Macdonald, C., Ounis, I.: Inferring conceptual relationships to improve medical records search. In: Proceedings of the 10th conference on open research areas in information retrieval. pp. 1–8. LE CENTRE DE HAUTES ETUDES INTERNATIONALES D'INFORMATIQUE DOCUMENTAIRE (2013)

[65] Limsopatham, N., Macdonald, C., Ounis, I.: A task-specific query and document representation for medical records search. In: European Conference on Information Retrieval. pp. 747–751. Springer (2013)

[66] Lipani, A., Roelleke, T., Lupu, M., Hanbury, A.: A systematic approach to normalization in probabilistic models. Inf. Retr. Journal **21**(6), 565–596 (2018). https://doi.org/10.1007/s10791-018-9334-1, https://doi.org/10.1007/s10791-018-9334-1

[67] Liu, B.: Sentiment analysis and opinion mining. Synthesis lectures on human language technologies **5**(1), 1–167 (2012)

[68] Liu, B., et al.: Sentiment analysis and subjectivity. Handbook of natural language processing **2**(2010), 627–666 (2010)

[69] Maas, A.L., Daly, R.E., Pham, P.T., Huang, D., Ng, A.Y., Potts, C.: Learning word vectors for sentiment analysis. In: Proceedings of the 49th Annual Meeting of the Association for Computational Linguistics: Human Language Technologies. pp. 142–150. Association for Computational Linguistics, Portland, Oregon, USA (June 2011), http://www.aclweb.org/anthology/P11-1015

[70] Madsen, R.E., Kauchak, D., Elkan, C.: Modeling word burstiness using the dirichlet distribution. In: Proceedings of the 22nd international conference on Machine learning. pp. 545–552 (2005)

[71] Margulis, E.: N-Poisson document modelling. In: Belkin, N., Ingwersen, P., Pejtersen, M. (eds.) Proceedings of the Fifteenth Annual

International ACM SIGIR Conference on Research and Development in Information Retrieval. pp. 177–189. ACM, New York (1992)

[72] Meghini, C., Sebastiani, F., Straccia, U., Thanos, C.: A model of information retrieval based on a terminological logic. In: Korfhage, R., Rasmussen, E., Willett, P. (eds.) Proceedings of the Sixteenth Annual International ACM SIGIR Conference on Research and Development in Information Retrieval. pp. 298–308. ACM, New York (1993)

[73] Meghini, C., Rabitti, F., Thanos, C.: Conceptual modeling of multimedia documents. Computer **24**(10), 23–30 (1991)

[74] Meij, E., Trieschnigg, D., De Rijke, M., Kraaij, W.: Conceptual language models for domain-specific retrieval. Information Processing & Management **46**(4), 448–469 (2010)

[75] Miller, G.A., Beckwith, R., Fellbaum, C., Gross, D., Miller, K.J.: Introduction to wordnet: An on-line lexical database. International journal of lexicography **3**(4), 235–244 (1990)

[76] Minka, T.: Estimating a dirichlet distribution (2000)

[77] Nie, J.: Towards a probabilistic modal logic for semantic-based information retrieval. In: Belkin, N., Ingwersen, P., Pejtersen, M. (eds.) Proceedings of the Fifteenth Annual International ACM SIGIR Conference on Research and Development in Information Retrieval. pp. 140–151. ACM, New York (1992)

[78] Paik, J.H.: A novel TF-IDF weighting scheme for effective ranking. In: Jones, G.J.F., Sheridan, P., Kelly, D., de Rijke, M., Sakai, T. (eds.) The 36th International ACM SIGIR conference on research and development in Information Retrieval, SIGIR '13, Dublin, Ireland - July 28 - August 01, 2013. pp. 343–352. ACM (2013)

[79] Pang, B., Lee, L., Vaithyanathan, S.: Thumbs up?: sentiment classification using machine learning techniques. In: Proceedings of the ACL-02 conference on Empirical methods in natural language processing-Volume 10. pp. 79–86. Association for Computational Linguistics (2002)

[80] Patro, S.G.K., Mishra, B.K., Panda, S.K., Kumar, R., Long, H.V., Taniar, D., Priyadarshini, I.: A hybrid action-related k-nearest neighbour (har-knn) approach for recommendation systems. IEEE Access **8**, 90978–90991 (2020)

[81] Pedregosa, F., Varoquaux, G., Gramfort, A., Michel, V., Thirion, B., Grisel, O., Blondel, M., Prettenhofer, P., Weiss, R., Dubourg, V., et al.: Scikit-learn: Machine learning in python. the Journal of machine Learning research **12**, 2825–2830 (2011)

[82] Pennebaker, J.W., Francis, M.E., Booth, R.J.: Linguistic inquiry and word count: Liwc 2001. Mahway: Lawrence Erlbaum Associates **71**(2001), 2001 (2001)

[83] Pollard, S., Biermann, A.W.: A measure of semantic complexity for natural language systems. In: NAACL-ANLP 2000 Workshop: Syntactic and Semantic Complexity in Natural Language Processing Systems. pp. 42–46. Association for Computational Linguistics (2000)

[84] Ponte, J.M., Croft, W.B.: A language modeling approach to information retrieval. In: Proceedings of the 21st annual international ACM SIGIR conference on Research and development in information retrieval. pp. 275–281 (1998)

[85] Robertson, S.E., Walker, S.: Some simple effective approximations to the 2-Poisson model for probabilistic weighted retrieval. pp. 232–241 (1994)

[86] Robertson, S.: Understanding inverse document frequency: on theoretical arguments for idf. Journal of documentation **60**(5), 503–520 (2004)

[87] Robertson, S., Hull, D.A.: The trec-9 filtering track final report. In: TREC. vol. 10, pp. 344250–344253. Citeseer (2000)

[88] Roelleke, T.: Information retrieval models: foundations and relationships. Synthesis Lectures on Information Concepts, Retrieval, and Services **5**(3), 1–163 (2013)

[89] Roelleke, T., Kaltenbrunner, A., Baeza-Yates, R.: Harmony assumptions in information retrieval and social networks. The Computer Journal **58**(11), 2982–2999 (2015)

[90] Roelleke, T., Wang, J.: A parallel derivation of probabilistic information retrieval models. In: ACM SIGIR. pp. 107–114. Seattle, USA (2006)

[91] Sanz-Cruzado, J., Castells, P.: Information retrieval models for contact recommendation in social networks. In: European Conference on Information Retrieval. pp. 148–163. Springer (2019)

[92] Sanz-Cruzado, J., Castells, P., Macdonald, C., Ounis, I.: Effective contact recommendation in social networks by adaptation of information retrieval models. Information Processing & Management **57**(5), 102285 (2020)

[93] Scholer, F., Williams, H.E., Turpin, A.: Query association surrogates for web search. Journal of the American Society for Information Science and Technology **55**(7), 637–650 (2004)

[94] Shen, W., Nie, J.Y.: Is concept mapping useful for biomedical information retrieval? In: International conference of the cross-language evaluation forum for European languages. pp. 281–286. Springer (2015)

[95] Sil, A., Yates, A.: Re-ranking for joint named-entity recognition and linking. In: Proceedings of the 22nd ACM international conference on Conference on information & knowledge management. pp. 2369–2374. ACM (2013)

[96] Singh, V.K., Piryani, R., Uddin, A., Waila, P.: Sentiment analysis of movie reviews: A new feature-based heuristic for aspect-level sentiment classification. In: 2013 International Mutli-Conference on Automation, Computing, Communication, Control and Compressed Sensing (iMac4s). pp. 712–717. IEEE (2013)

[97] Sondak, M., Shtok, A., Kurland, O.: Estimating query representativeness for query-performance prediction. In: Proceedings of the 36th international ACM SIGIR conference on Research and development in information retrieval. pp. 853–856. ACM (2013)

[98] Strang, G.: The fundamental theorem of linear algebra. The American Mathematical Monthly **100**(9), 848–855 (1993)

[99] Suominen, H., Salanterä, S., Velupillai, S., Chapman, W.W., Savova, G., Elhadad, N., Pradhan, S., South, B.R., Mowery, D.L., Jones, G.J., et al.: Overview of the share/clef ehealth evaluation lab 2013. In: International Conference of the Cross-Language Evaluation Forum for European Languages. pp. 212–231. Springer (2013)

[100] Tan, S., Duan, Z., Zhao, S., Chen, J., Zhang, Y.: Improved reviewer assignment based on both word and semantic features. Information Retrieval Journal **24**(3), 175–204 (2021)

[101] Turney, P.D., Littman, M.L.: Measuring praise and criticism: Inference of semantic orientation from association. ACM Transactions on Information Systems (TOIS) **21**(4), 315–346 (2003)

[102] Valcarce, D.: Exploring statistical language models for recommender systems. In: Proceedings of the 9th ACM Conference on Recommender Systems. pp. 375–378 (2015)

[103] Valcarce, D., Bellogín, A., Parapar, J., Castells, P.: Assessing ranking metrics in top-n recommendation. Information Retrieval Journal **23**(4), 411–448 (2020)

[104] Valcarce, D., Parapar, J., Barreiro, Á.: A study of smoothing methods for relevance-based language modelling of recommender systems. In: European Conference on Information Retrieval. pp. 346–351. Springer (2015)

[105] Valcarce, D., Parapar, J., Barreiro, Á.: Axiomatic analysis of language modelling of recommender systems. International Journal of Uncertainty, Fuzziness and Knowledge-Based Systems **25**(Suppl. 2), 113–127 (2017)

[106] Van Zwol, R., Van Loosbroek, T.: Effective use of semantic structure in xml retrieval. In: European Conference on Information Retrieval. pp. 621–628. Springer (2007)

[107] Voorhees, E.M.: Overview of TREC 2004. In: Voorhees, E.M., Buckland, L.P. (eds.) Proceedings of the Thirteenth Text REtrieval Conference, TREC 2004, Gaithersburg, Maryland, USA, November 16-19, 2004. NIST Special Publication, vol. 500-261. National Institute of Standards and Technology (NIST) (2004), http://trec.nist.gov/pubs/trec13/papers/OVERVIEW13.pdf

[108] Voorhees, E.M., Hersh, W.R.: Overview of the TREC 2012 medical records track. In: Voorhees, E.M., Buckland, L.P. (eds.) Proceedings of The Twenty-First Text REtrieval Conference, TREC 2012, Gaithersburg, Maryland, USA, November 6-9, 2012. NIST Special Publication, vol. 500-298. National Institute of Standards and Technology (NIST) (2012), http://trec.nist.gov/pubs/trec21/papers/MED12OVERVIEW.pdf

[109] Wang, C., Akella, R.: Concept-based relevance models for medical and semantic information retrieval. In: Proceedings of the 24th ACM International on Conference on Information and Knowledge Management. pp. 173–182 (2015)

[110] Wang, J., De Vries, A.P., Reinders, M.J.: Unified relevance models for rating prediction in collaborative filtering. ACM Transactions on Information Systems (TOIS) **26**(3), 1–42 (2008)

[111] Wang, J., Robertson, S., de Vries, A.P., Reinders, M.J.: Probabilistic relevance ranking for collaborative filtering. Information Retrieval **11**(6), 477–497 (2008)

[112] Wang, Y.X., Zhang, Y.J.: Nonnegative matrix factorization: A comprehensive review. IEEE Transactions on Knowledge and Data Engineering **25**(6), 1336–1353 (2012)

[113] Wawre, S.V., Deshmukh, S.N.: Sentiment classification using machine learning techniques. International Journal of Science and Research (IJSR) **5**(4), 819–821 (2016)

[114] Wilson, T., Wiebe, J., Hwa, R.: Just how mad are you? finding strong and weak opinion clauses. In: aaai. vol. 4, pp. 761–769 (2004)

[115] Xiong, C., Dai, Z., Callan, J., Liu, Z., Power, R.: End-to-end neural ad-hoc ranking with kernel pooling. In: Proceedings of the 40th International ACM SIGIR Conference on Research and Development in Information Retrieval. pp. 55–64 (2017)

[116] Xu, Z., Akella, R.: A new probabilistic retrieval model based on the dirichlet compound multinomial distribution. In: Proceedings of the 31st Annual International ACM SIGIR Conference on Research and Development in Information Retrieval. p. 427–434 (2008)

[117] Zendel, O., Shtok, A., Raiber, F., Kurland, O., Culpepper, J.S.: Information needs, queries, and query performance prediction. In: Proceedings of the 42nd International ACM SIGIR Conference on Research and Development in Information Retrieval. pp. 395–404 (2019)

[118] Zhai, C.: Statistical Language Models for Information Retrieval. Morgan & Claypool Publishers (2009)

[119] Zhai, C., Lafferty, J.: A study of smoothing methods for language models applied to information retrieval. ACM Transactions on Information Systems (TOIS) **22**(2), 179–214 (2004)

[120] Zhang, W., Yu, C., Meng, W.: Opinion retrieval from blogs. In: Proceedings of the conference on information and knowledge management. ACM (2007)

[121] Zhao, Y., Scholer, F., Tsegay, Y.: Effective pre-retrieval query performance prediction using similarity and variability evidence. In: European conference on information retrieval. pp. 52–64. Springer (2008)

[122] Ziegler, C.N., McNee, S.M., Konstan, J.A., Lausen, G.: Improving recommendation lists through topic diversification. In: Proceedings of the 14th international conference on World Wide Web. pp. 22–32 (2005)

Appendix A

Tables

Number	Topic	Analysis		
		Length	**Language**	**Need**
1	Find articles about Ferroportin-1, an iron transporter, in humans.	9	Declarative	EI
2	What is the time course of gene expression in the murine developing kidney.	13	Interrogative	PM
3	What mouse genes are specific to the kidney.	8	Interrogative	M
4	Articles are relevant if they describe methods for subcellular fractionation of nuclei.	12	Declarative	EI
5	Find articles about function of FancD2.	6	Declarative	EI
6	Genes and proteins (pathways) common to DNA repair, oxidative diseases, skin-carcinogenesis, and UV-carcinogenesis.	13	Declarative	M

7	Find articles about the function of mutY in humans.	9	Declarative	EI
8	Find articles about the role of NEIL1 in repair of DNA.	11	Declarative	EI
9	Find articles describing genes that are regulated by the signal transducing molecule Smad4.	13	Declarative	EI
10	Documents regarding the role of TGFB in angiogenesis in skin with respect to homeostasis and development.	16	Declarative	EI
11	Documents regarding TGFB expression or regulation in the cancers called HNSCC.	11	Declarative	EI
12	Find information on role of ATPase in Apoptosis.	8	Declarative	PM
13	Properties of Gis4 with respect to cell cycle and the metabolism.	11	Declarative	PM
14	Do p63 and p73 cause cell cycle arrest or apoptosis related to the dna damage.	15	Interrogative	Y/NO
15	Find all reports describing proteins related to peptidoglycan recognition of the mouses.	12	Declarative	EI
16	Retrieval of information regarding the role of BUB2 and BFA1 in cytokinesis in the yeast.	15	Declarative	PM

17	Studies that investigate similarities in morphological changes among apoptosis and autophagy processes.	12	Declarative	EI
18	Documents containing the sequences and phenotypes of E. coli gyrA mutations.	10	Declarative	EI
19	Documents identifying genes that are regulated by a gene called Nkx family members.	13	Declarative	EI
20	Reports that provide possible links between neurofibromatosis and TOR signaling	10	Declarative	EI
21	Find reports that describe xenograft models of cancers.	8	Declarative	EI
22	Identify research on mutant mouse strains and factors which increase susceptibility to infection by Histoplasma capsulatum.	16	Declarative	PM
23	Articles reporting experiments allowing annotation of the products based on genes related the cryptococcus.	14	Declarative	EI
24	What is the function of proteins containing WD40 repeats.	9	Interrogative	PM
25	What research is being done on the enzyme which is called peptide amidating or PAM.	15	Interrogative	PM

26	Information concerning genetic loci that are associated with increased risk of the stroke. Such as the apolipoprotein that called E4 or factor V mutations.	24	Declarative	PM
27	Identify genes as potential genetic risk factors candidates for causing hypertension.	11	Declarative	M
28	To identify the antigens expressed by epithelial cells of lung and the antibodies available.	14	Declarative	M
29	What are the phenotypes that have been described resulting from mutations in the Cystic Fibrosis conductance regulator gene.	18	Interrogative	M
30	What genes show altered behavior due to chromosomal rearrangements.	9	Interrogative	M
31	Studies of Sleeping Beauty transposons.	5	Declarative	EI
32	Research the gene of human named BCL-2 to determine if there are antagonists and inhibitors inside of a cell.	19	Declarative	Y/NO
33	What is the focus of studies involving the members of the gene of human from UNC family.	17	Interrogative	EI
34	Find reports and genes that are glyphosate tolerance sequences in the literature.	12	Declarative	EI

35	Find research on improving protein expressions at low temperature in Escherichia coli bacteria.	13	Declarative	PM

Table A.1: Actual queries and the analysis (TREC-2004).

Number	Topic	Analysis		
		Length	**Language**	**Need**
1	Describe the procedure or methods for how to open up a cell through a process called electroporation.	17	Declarative	DP
2	Describe the procedure or methods for exact reactions that take place when you do glutathione S-transferase (GST) cleavage during affinity chromatography.	21	Declarative	DP
3	Describe the procedure or methods for different quantities of different components to use when pouring a gel to make it more or less porous.	24	Declarative	DP
4	Describe the procedure or methods for green fluorescent protein (GFP) tagged proteins to do experiments with tagged proteins.	18	Declarative	DP
5	Describe the procedure or methods for how to do a microsomal budding assay, i.e., budding of vesicles from microsomes in vitro.	20	Declarative	DP
6	Describe the procedure or methods for purification of rat IgM.	10	Declarative	DP
7	Describe the procedure or methods for chromatin IP (Immuno Precipitations) to isolate proteins that are bound to DNA in order to precipitate the proteins out of the DNA.	27	Declarative	DP

8	Describe the procedure or methods for normalization procedures that are used for microarray data.	14	Declarative	DP
9	Describe the procedure or methods for identifying in vivo protein-protein interactions in time and space in the living cell.	19	Declarative	DP
10	Describe the procedure or methods for fluorogenic 5'-nuclease assay.	9	Declarative	DP
11	Provide information about the role of Interferon-beta gene in the disease Multiple Sclerosis.	13	Declarative	PI
12	Provide information about the role of PRNP gene in the disease Mad Cow Disease.	14	Declarative	PI
13	Provide information about the role of APC (adenomatous polyposis coli) gene in the disease Colon Cancer.	14	Declarative	PI
14	Provide information about the role of Nurr-77 gene in the disease called the parkinson disease.	15	Declarative	PI
15	Provide information about the role of Insulin receptor gene in the cancer.	12	Declarative	PI
16	Provide information about the role of the gene Apolipoprotein E (ApoE) in the disease called the Alzheimer's Disease.	18	Declarative	PI

17	Provide information about the role of Transforming growth factor-beta1 (TGF-beta1) gene in the disease that is called cerebral amyloid angiopathy.	20	Declarative	PI
18	Provide information about the role of GSTM1 gene in the breast cancer.	12	Declarative	PI
19	Provide information on the role of nucleoside diphosphate kinase or (NM23) genes in the process of the tumor progression.	19	Declarative	PI
20	Provide information on the role of APC (adenomatous polyposis coli) gene in the process of the assembly of the actin.	18	Declarative	PI
21	Provide information on the role of casein kinase II gene in the process of the assembly of the ribosome.	19	Declarative	PI
22	Provide information on the role of P53 genes in the process of apoptosis.	13	Declarative	PI
23	Provide information on the role of alpha7 nicotinic receptor gene subunit gene in the process of the ethanol metabolism.	19	Declarative	PI
24	Provide information on the role of Interferon-beta gene in the process of viral entry into host cell.	17	Declarative	PI
25	Provide information about the genes called BRCA1 regulation of the ubiquitin in the cancer.	14	Declarative	PI

26	Provide information about alpha7 nicotinic receptor gene and ApoE gene the neurotoxic effects of the ethanol.	16	Declarative	PI
27	Provide information about HNF4 and COUP-TF I genes in the suppression in the function of the liver.	17	Declarative	PI
28	Provide information about Ret and GDNF genes in the development of the kidney.	13	Declarative	PI
29	Provide information about Mutations of presenilin-1 gene and the biological impact in Alzheimer's disease.	14	Declarative	PI
30	Provide information about the Mutation of type 1 of the familial hemiplegic migraine that is known as (FHM1) and the neuronal Ca2+ influx in the hippocampal neurons.	26	Declarative	PI

Table A.2: Actual queries and the analysis (TREC-2005).

Index