Using IMRaD Structure Features in Information Retrieval Ranking Functions

Thomas Mauerhofer

November 26, 2020

Institute of Interactive Systems and Data Science

Introduction

Motivation

Science

- growing fast
- number of paper submissions increases
- finding relevant information is getting more time-consuming

Example: Top-Tier Computer Vision Conferences

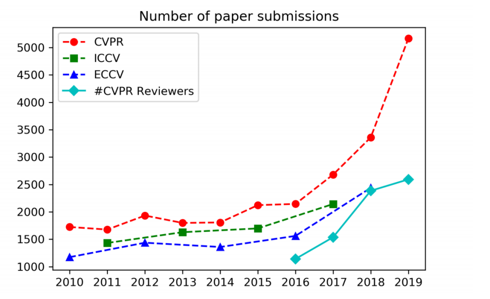


Figure Source: Deep Paper Gestalt

Motivation

Search Engine

- filter data
- reduce time that is required to search thought different information sources
- usage of explicit and implicit information

Motivation

Improve Literature Search Process

- reduce the amount of non relevant scientific articles
- scientific articles share a common structure (IMRaD)

Example[1]

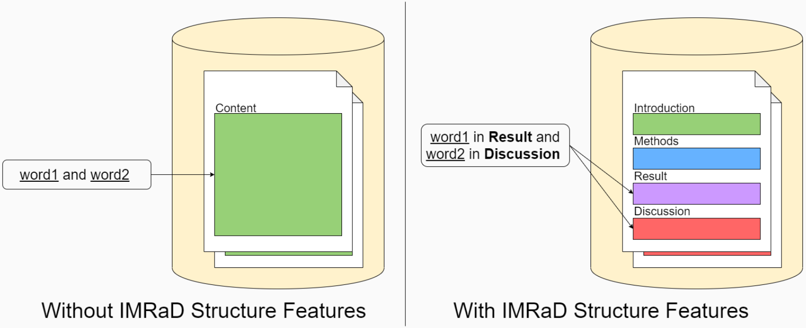
Section Name	IMRaD Type
Introduction	Introduction
Related work	Methods
Extracting contiguous text blocks	Methods
Evaluation	Results
Discussion	Discussion

[1] Section Titles of Klampfl et al. [3] are used

Research Question

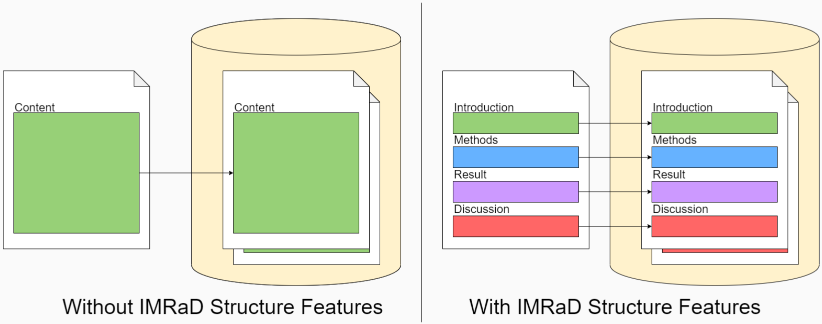
Is it possible to improve the search result quality by using IMRaD structure features?

1. Does the quality improve for explicit search using queries?



Research Question

2. Does the quality improve for implicit search using scientific articles?
3. Does the quality improve if only a single section is used for searching?

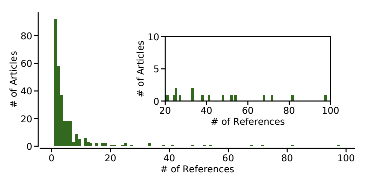


Materials and Method

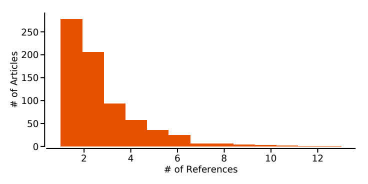
Dataset

Scientific Article Dataset

- consists of 821 articles
- generated citation network
    - References (Links): 1,716



In-degree Distribution - Mean 5.9



Out-degree Distribution - Mean 2.4

Dataset

Added IMRaD Structure Information

- classify the IMRaD types with keyword detection in section titles
- **Related Work** as additional IMRaD type called **Background**
- **Methods** have less common keywords

IMRaD Type	Section Title Term	# Paper	Percent
Introduction	Introduction	822	100%
Background	Related Work	465	56.57%
Methods	Method, Model, Approach	312	37.96%
Result	Experience, Result, Evaluation	687	83.58%
Discussion	Conclusion, Discussion, Future Work	773	94.04%

Implementation

Desgin Goals

- various common ranking algorithms should be comparable
- works with unstructured as well as structured data

Technologies

- **Backend Implementation:** Python
- **Database:** MongoDB
- **Web-Framework:** Flask
- **Frontend Implementation:** Bootstrap/jQuery

Information Retrieval Model

Defined as Quadruple $[\mathsf{D}, \mathsf{Q}, \mathcal{F}, \mathcal{R}(q_i, d_j)]$ [5]

- D ...representation of the documents in a collection
- Q ...representation of the user information needs (i.e., queries)
- $\mathcal{R}(q_i, d_j)$ ...raking function
- $\mathcal{F}$ ...framework

Example

- documents D are represented as Bags of Words
- queries Q are represented as sets
- $\mathcal{R}(q_i, d_j) = \sum_{t \in q_i} TF(d_j, t)$

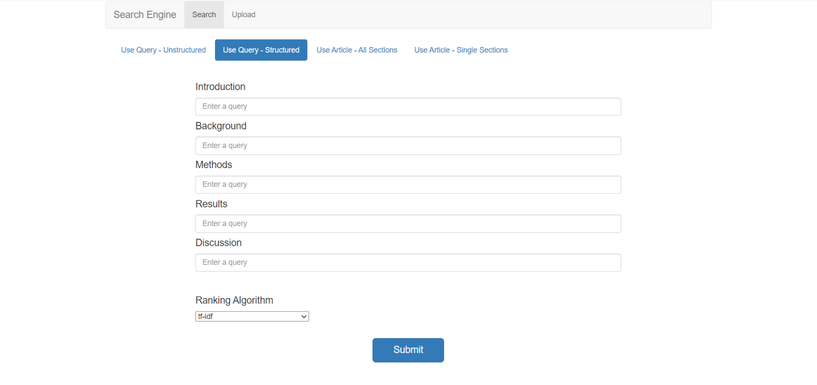
Information Retrieval Model

Model Design

- each document consists of 6 Bag of Words
  - one for unstructured retrieval, and one for each IMRaD type
- each query consists of 6 sets
- structured retrieval ranking formula:

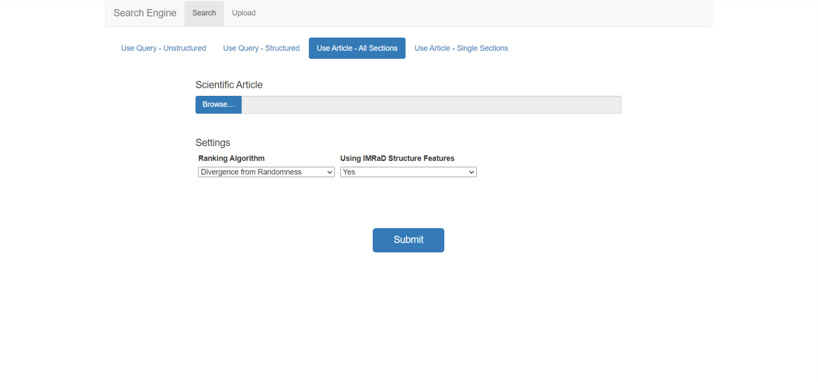
$$sim(d_j, q) = \frac{1}{|\text{IMRaD-TYPES}|} \times \sum_{k \in \text{IMRaD-TYPES}} sim(d_{j,k}, q_k)$$

Search with User Query

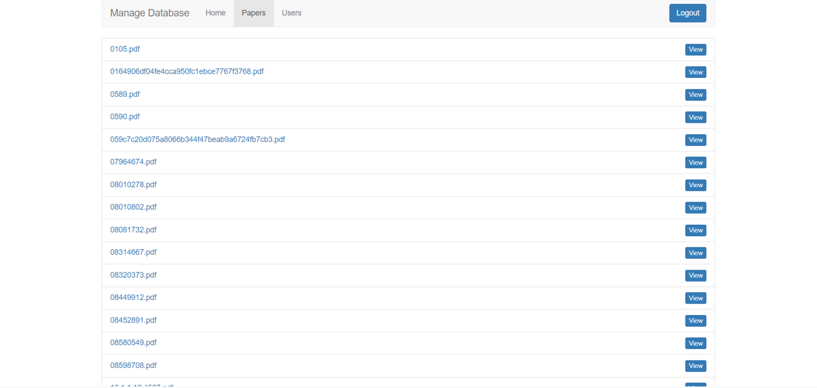


User Interface

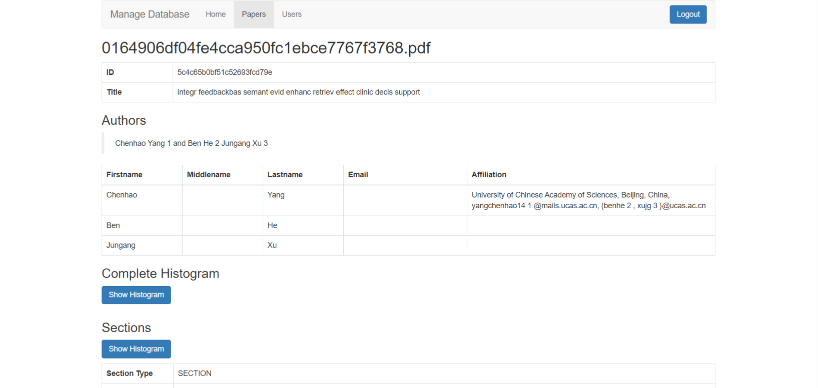
Search with Scientific Article



Admin Panel - Overview of all Articles



Admin Panel - Article Details



Results and Discussion

1. Experiment - Evaluate based on User Queries

Experimental setup

- generated Word N-Gramms with citations in the articles
- IMRaD type is defined by the section the citation occurs
- query length from 2 to 14

Results

Using IMRaD Structure Features		Term Frequency	TF-IDF	Ranked Boolean Retrieval	BM25	Divergence from Randomness
No	**Best Accuracy**	0.1966	0.2199	0.1921	0.1207	0.0498
	Query Length	11	11	11	14	2
Yes	**Best Accuracy**	0.1293	0.1642	0.1015	0.1058	0.0379
	Query Length	12	12	9	13	2

→ IMRaD Structure Features does not improve search results based on our assumptions

2. Experiment - Evaluate based on Scientific Articles

Experimental Setup

- relevant documents based on referenced articles

Results

Using IMRaD Structure Features		Term Frequency	TF-IDF	Ranked Boolean Retrieval	BM25	Divergence from Randomness
No	**Accuracy**	0.1186	0.1163	0.0466	0.0554	0.0137
Yes	**Accuracy**	0.1463	0.1613	0.0506	0.0882	0.0137

→ IMRaD Structure Features improve search results when scientific articles are used

3. Experiment - Evaluate based on single Sections

Experimental setup

- only structured with usage of scientific articles
- one IMRaD type is used in query (Input Area) and in documents (Search Area)

Results (represented using TF-IDF)

Search Area

Input Area: Section	Introduction	Background	Methods	Results	Discussion
Introduction	0.1242	0.1226	0.1095	0.1092	0.1049
Background	0.1454	0.1249	0.1331	0.1255	0.1106
Methods	0.0947	0.0857	0.1017	0.0897	0.0668
Results	0.0877	0.0783	0.0815	0.0783	0.0631
Discussion	0.1188	0.1078	0.0957	0.0914	0.084

→ Introduction and Background tend to contain more relevant information

Summary

Results Overview

	Term Frequency	TF-IDF	Ranked Boolean Retrieval	BM25	Divergence from Randomness
Accuracies of 1. Experiment without IMRaD Structure Features	0.1966	0.2199	0.1921	0.1207	0.0498
Accuracies of 2. Experiment with IMRaD Structure Features	0.1463	0.1613	0.0506	0.0882	0.0137
Accuracies of 3. Experiment with IMRaD Structure Features	-	0.1454	-	-	-

→ first two experiments cover different requirements of a user

1. breadth-first search and covers the initial search process
2. depth-first search and covers the specific search of literature

Results Overview

	Term Frequency	TF-IDF	Ranked Boolean Retrieval	BM25	Divergence from Randomness
Accuracies of 1. Experiment without IMRaD Structure Features	0.1966	0.2199	0.1921	0.1207	0.0498
Accuracies of 2. Experiment with IMRaD Structure Features	0.1463	0.1613	0.0506	0.0882	0.0137
Accuracies of 3. Experiment with IMRaD Structure Features	-	0.1454	-	-	-

→ first two experiments cover different requirements of a user

1. breadth-first search and covers the initial search process
2. depth-first search and covers the specific search of literature

→ for the 3. experiment queries and documents with similar performance significant smaller compared to the 2. experiment

References i

Gianni Amati and C. J. van Rijsbergen. “Probabilistic models of information retrieval based on measuring the divergence from randomness.”. In: *ACM Trans. Inf. Syst.* 20.4 (2002), pp. 357–389.

Karen Spärck Jones. “A statistical interpretation of term specificity and its application in retrieval”. In: *Journal of Documentation* 28.1 (1972).

Stefan Klampfl et al. “Unsupervised document structure analysis of digital scientific articles”. In: *Int. J. on Digital Libraries* 14.3-4 (2014), pp. 83–99.

Christopher D. Manning, Prabhakar Raghavan, and Hinrich Schütze. *Introduction to Information Retrieval*. Cambridge University Press, 2008.

References ii

- Berthier Ribeiro-Neto and Ricardo Baeza-Yates. *Modern Information Retrieval*. ACM Press / Addison-Wesley, 1999.
- Stephen E. Robertson et al. “Okapi at TREC-2.”. In: *TREC*. Ed. by Donna K. Harman. Vol. 500-215. NIST Special Publication. National Institute of Standards and Technology (NIST), 1993, pp. 21–34.
- Stephen E. Robertson et al. “Okapi at TREC-3.”. In: *TREC*. Ed. by Donna K. Harman. Vol. 500-225. NIST Special Publication. National Institute of Standards and Technology (NIST), 1994, pp. 109–126.
- Stephen E. Robertson et al. “Okapi at TREC.”. In: *TREC*. Ed. by Donna K. Harman. Vol. 500-207. NIST Special Publication. National Institute of Standards and Technology (NIST), 1992, pp. 21–30.

References iii

G. Salton and C. S. Yang. "On the specification of term values in automatic indexing.". In: *Journal of Documentation.* 29.4 (1973), pp. 351–372.

Ranking Functions

Term Frequency - Inverted Document Frequency (TF-IDF) [2, 9]

$$sim(d_j, q) = f_{i,j} \times \log \frac{N}{n_i}$$

- includes the importance of a term with respect to the whole document collection
- multiple variants of TF-IDF

BM25

$$\mathcal{B}_{i,j} = \frac{(K_1 + 1) f_{i,j}}{K_1 \left[(1 - b) + b \frac{len(d_j)}{avg\_doclen} \right] + f_{i,j}}$$

$$sim_{\mathrm{BM25}}(d_j, q) \sim \sum_{k_i \in q \wedge k_i \in d_j} \mathcal{B}_{i,j} \times \log \left(\frac{N - n_i + 0.5}{n_i + 0.5} \right)$$

- result of several experiments by Robertson et al. [6, 7, 8]
- combination of BM15 and BM11
  - BM11 additionally uses document length normalization
  - parameter to define the influence of the 2 terms

Ranking Functions

Divergence from Randomness [1]

$$w_{i,j} = (-\log P(k_i|C)) \times (1 - P(k_i|d_j))$$
$$R(d_j, q) = \sum_{k_i \in q} f_{i,q} \times w_{i,j}$$

- based on 2 assumptions:
  1. amount of information for a term over the whole document collection: $-\log P(k_i|C)$
  2. amount of information for a term being in a complementary term distribution: $1 - P(k_i|d_j)$

$$-\log P(k_i|C) \approx f_{i,j} \log\left(\frac{f_{i,j}}{\lambda_i}\right) + \left(\lambda_i + \frac{1}{12 f_{i,j} + 1} - f_{i,j}\right) \log e + \frac{1}{2} \log(2\pi f_{i,j})$$
$$1 - P(k_i|d_j) = \frac{1}{f_{i,j} + 1}$$

Ranking Functions

Ranked Boolean Retrieval [4]

$$\sum_{i=1}^{l} g_i s_i$$

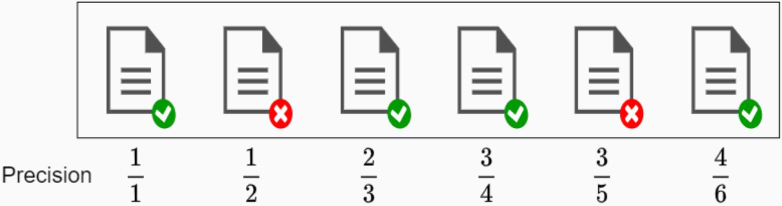
- documents are divided into zones
- based on zone scores
- apply zone score to result when a term occurs in zone

Evaluation of Ranking Algorithms

Mean Average Precision

- evaluate search result (ordered ranked lists)
- calculate average precision based on a set with relevant documents

Example - Average Precision of a single query



$$\rightarrow AP_i = \frac{\sum_{k=1}^{|R_i|} P(R_i[k])}{|R_i|} = \frac{(\frac{37}{12})}{4} \approx 0.77$$

Word N-Gramm

Generate Test Queries

- **Assumption:** citations describe the content of referenced articles
- used Word N-Gramm
- added additional information about referenced article and IMRaD type of the section

Example: N = 6

”Information Retrieval Systems are important to reduce research time [1]”

→ ”Information Retrieval Systems important reduce research”
”Retrieval Systems important reduce research time”

Future Work

- calculate similarities based on article clusters
- reevaluate the first experiment with different assumption about the occurrence of the query terms