ingenieur
wissenschaften
htw saar
Information Retrieval
(WS 2018/2019)

Klaus Berberich
(klaus.berberich@htw-saar.de)
0. Organization
Lectures / Exercises

- **Lectures and exercises** will take place on
  - **Monday** 08:15– 09:45 in room 7110
  - **Thursday** 08:15 – 09:45 in room 7110

- The **detailed schedule** on when exercises will be discussed will be available on the **course website**
Website

- There is a **website** with all details about this course
  - [https://swl.htwsaar.de/lehre/ws18/ir/](https://swl.htwsaar.de/lehre/ws18/ir/)

- On the website you will find **slides**, **exercise sheets**, and the **datasets** for the programming assignments

- Some areas of the website will be **password protected**
  - Username: **ir**
  - Password: **7110**
Exercises / Programming Assignments

- There will be **four exercises**, with problems that you can solve on paper, and **four programming** assignments, for which you need to write code.

- In the **programming assignments**, we will develop our own little search engine and evaluate how well it works.

- It is **up to you** whether you **hand in a solution** to these.
Bonus Points

- By submitting solutions to the exercises and programming assignments, you can obtain up to 30 bonus points (percent)

- These are the rules for obtaining bonus points
  - you can submit by e-mail in teams of up to three people
  - you have to submit by the deadline on the exercise sheet
  - you have to pass the exam at the end of the lecture period
    - 50% in exam and 30 bonus points = 80% in exam (i.e., 2.0)
    - 20% in exam and 30 bonus points = 20% in exam (i.e., you fail)
  - bonus points are only valid this semester
Exam

- There will be a **written exam** in the last session of this course, i.e., February 7\textsuperscript{th} from 08:15 until 09:45

- The exam will take **90 minutes** and you are allowed to bring **three handwritten sheets of DIN-A4 paper** with your own notes as well as a **non-programmable pocket calculator**
Literature

  [Online]

  [Online]
Agenda

- 1. Introduction
- 2. Natural Language Preprocessing
- 3. Retrieval Models
- 4. IR-System Implementation
- 5. Evaluation
- 6. Web Search
- 7. Semantic Search
1. Introduction
Information Retrieval is Everywhere
Information Retrieval (IR) is finding material (usually documents) of an unstructured nature (usually text) that satisfies an information need from within large collections (usually stored on computers)

Manning et al. [1]
What is Information Retrieval?

Information Retrieval (IR) is finding material (usually documents) of an unstructured nature (usually text) that satisfies an information need from within large collections (usually stored on computers)

Manning et al. [1]

- Information Retrieval (IR) is about finding content, e.g.:
  - articles (e.g., scientific reports, newspaper articles)
  - office documents (e.g., letters or spreadsheets)
  - multimedia content (e.g., images or videos)
  - web pages, e-mails, social media profiles, etc.
What is Information Retrieval?

Information Retrieval (IR) is finding material (usually documents) of an unstructured nature (usually text) that satisfies an information need from within large collections (usually stored on computers)

Manning et al. [1]

- Contents have no or little predefined structure (in contrast to tuples in relational databases)
  - simple text documents in natural language
  - HTML documents with some markup (e.g., for headers)
  - semi-structured documents (e.g., XML or JSON)
What is Information Retrieval?

Information Retrieval (IR) is finding material (usually documents) of an unstructured nature (usually text) that satisfies an information need from within large collections (usually stored on computers).

- IR seeks to satisfy an information need of a human user
  - information need is often vague (e.g., learn about robotics) and expressed as one or multiple queries (e.g., introduction robotics)
  - only the human user can say whether a document is relevant

Manning et al. [1]
What is Information Retrieval?

Information Retrieval (IR) is finding material (usually documents) of an unstructured nature (usually text) that satisfies an information need from within large collections (usually stored on computers)

- Documents collections can be very large and dynamic
  - 100,000 documents on a desktop computer
  - 10,000,000 articles in a newspaper archive
  - >> 1,000,000,000,000 web pages on the World Wide Web
  - 500,000,000 tweets per minute on Twitter

Manning et al. [1]
Information Retrieval is Interdisciplinary
Historical Background

- **Libraries** (dating back to 3000 B.C.)
  - organize contents in **catalogues** according to **author**, **publication year** or **keywords**
  - categorize content using a **classification scheme** (e.g., Dewey Decimal Classification)

- Vannevar Bush’s idea of a **MemEx** (1945)
  - serves as a **memory extender**
  - foresees **storage, cross-linking, and retrieval** of contents
Historical Background

- **SMART system** developed by Salton et al. (1960s)
  - full-text indexing and result ranking
  - brought user "in the loop" by asking for relevance feedback

- **TREC** and other benchmark initiatives (since 1990s)
  - reusable testbeds with documents, information needs, and relevance judgments
Historical Background

- **Google** (1998)
  - improved web search by making use of the link structure of the World Wide Web with their **PageRank** algorithm

- **Learning to Rank** (since 2000s)
  - observe **user behavior** (who clicks on what for which query) and use **Machine Learning** to rank documents in response to a query
  - progress in recent years through the use of **Deep Learning**, which avoids extensive feature engineering
Information Retrieval vs. Relational Databases

- **Relational databases**
  - data has a **predetermined schema with attributes** that have **precise semantics**

```
<table>
<thead>
<tr>
<th>CustomerId</th>
<th>FirstName</th>
<th>LastName</th>
</tr>
</thead>
<tbody>
<tr>
<td>13</td>
<td>Paul</td>
<td>McCartney</td>
</tr>
<tr>
<td>14</td>
<td>John</td>
<td>Lennon</td>
</tr>
</tbody>
</table>
```

- **SQL** provides a query language which allows expressing information needs with **precise semantics**

```sql
1   SELECT  *
2   FROM Customers
3   WHERE Name LIKE 'Mc%'
```
Information Retrieval vs. Relational Databases

- **Information Retrieval** (in contrast)
  - **data is mostly unstructured with no precise semantics**

  These technologies are used to develop machines that can substitute for humans and replicate human actions. Robots can be used in many situations and for lots of purposes, but today many are used in dangerous environments (including bomb detection and deactivation), manufacturing processes, or where humans cannot survive (e.g. in space). Robots can take on any form but some are made to resemble humans in appearance.

  Source: https://en.wikipedia.org/wiki/Robotics

- **information need** (e.g., *learn about robotics*) is often vague and expressed as a query (e.g., *introduction robotics*)
Important Questions

- How can we **preprocess natural language texts**, e.g., to merge different forms of the same word (e.g., house and houses) and detect sentence boundaries?
  
  (Chapter 2: Natural Language Preprocessing)

- How can we **formally model documents** and **queries** and decide which documents are most likely to satisfy the user’s information need?
  
  (Chapter 3: Retrieval Models)
Important Questions

- How can we quickly return results for a specific query? (Chapter 3: IR-System Implementation)

- How can we determine whether our IR system returns good results or whether it is better than another system? (Chapter 4: Evaluation)
Important Questions

- How can we leverage specifics of the World Wide Web such as markup and hyperlinks?
  (Chapter 5: Web Search)

- How can we make use of natural language processing techniques that better understand documents to improve the search experience?
  (Chapter 6: Semantic Search)
Preliminary Answer: Boolean Retrieval

- Convert documents into a **bag of words** by converting it to lower case and splitting at white spaces

These technologies are used to develop machines that can substitute for humans and replicate human actions. Robots can be used in many situations and for lots of purposes, but today many are used in dangerous environments (including bomb detection and deactivation), manufacturing processes, or where humans cannot survive (e.g. in space). Robots can take on any form but some are made to resemble humans in appearance.


- Queries are **Boolean expressions** over known words
  
  robots AND humans AND NOT (science AND AND fiction)
Preliminary Answer: Boolean Retrieval

- Documents seen as assignments to Boolean variables

- A document **matches** a query if the corresponding Boolean expression evaluates to True on its value assignment

- An obvious **shortcoming** of this simple retrieval model is that there is **no ranking** of result documents
Preliminary Answer: Inverted Index

- We can build an **index structure** to speed up retrieval of documents that contain a specific word.

- **Inverted index** (also: inverted file) consists of
  - **dictionary** with all known words
  - **posting lists** with details about word occurrences
Preliminary Answer: Inverted Index

- While conceptually simple, there are **a lot of details** to consider when implementing an inverted index
  - **which information** should be stored in the postings
  - how can we **compress the inverted index** to safe space, but also to speed up reading it from disk
  - how can we **efficiently process queries** using an inverted index, maybe without reading the entire posting lists
Preliminary Answer: Precision and Recall

- Let us assume that we know for all documents in our collection whether they are relevant or not to a query.

- We can distinguish between documents that are returned as results for the query by our system and documents that are not returned.
Preliminary Answer: Precision and Recall

- This gives us **four different categories of documents**
  
  - **Relevant Results**
    (true positives)
  
  - **Irrelevant Results**
    (false positives)
  
  - **Relevant Non-Results**
    (false negatives)
  
  - **Irrelevant Non-Results**
    (true negatives)

<table>
<thead>
<tr>
<th>Relevant Documents</th>
<th>Result</th>
</tr>
</thead>
<tbody>
<tr>
<td>tn</td>
<td>tn</td>
</tr>
<tr>
<td>tn</td>
<td>tn</td>
</tr>
<tr>
<td>tn</td>
<td>tn</td>
</tr>
<tr>
<td>tn</td>
<td>tp</td>
</tr>
<tr>
<td>tn</td>
<td>fp</td>
</tr>
<tr>
<td>tn</td>
<td>tn</td>
</tr>
</tbody>
</table>
Preliminary Answer: Precision and Recall

- **Precision** measures the system’s ability to return only relevant results

\[
\frac{\#tp}{\#tp + \#fp} = \frac{\# \text{ Relevant Results}}{\# \text{ Results}}
\]

- **Recall** measures the system’s ability to return all relevant results

\[
\frac{\#tp}{\#tp + \#fn} = \frac{\# \text{ Relevant Results}}{\# \text{ Relevant Documents}}
\]
Title

- Text