Empirical research in Information Retrieval

Djoerd Hiemstra

University of Twente

hiemstra@cs.utwente.nl

http://www.cs.utwente.nl/~hiemstra
Goal

- An introduction to doing real (measurable, repeatable) research
- Getting acquainted with the “TREC paradigm”
- Some hands-on experience
The empirical study

- Clearly laid out sequence of steps:
  1. hypothesis;
  2. method;
  3. results;
  4. conclusion.

- The environment must be carefully controlled if the results of an evaluation are to be trusted.
1. Your hypothesis

- System A outperforms system B on task C
  - e.g. Google’s Page Rank outperforms the vector space model with tf.idf weighting for searching home pages on the web
2. What method?

- Identify the techniques that will be used to establish the hypothesis.
  - choose data
  - choose suitable evaluation measures: assign values to results of your system
  - choose a statistical methodology: determine whether observed differences are significant

- The ability to repeat an experiment is a key feature of empirical research.
3. Results

- Compile and present the results.
  - Repeat a number of times
4. Conclusion

- Supporting the hypothesis...

- or rejecting it.
Summary

**DATA: BY THE NUMBERS**

- Number of years to get data: 3
- Number of years to interpret data: 2
- Number of years to write about data: 1.5
- Number of slides to present data: 1

![Image](www.phdcomics.com)
Empirical computer science research

- “3.7% of computer science journal papers use the laboratory experiment as the primary research method”

- ACM Transactions on Information Systems was the only journal in which comparative studies of systems (laboratory experiment) was used as the primary research method (14.3%)

The traditional IR experiment

- To start with you need
  - A system (or two)
  - A collection of documents / data
  - A collection of queries / requests

- Then you run your experiment
  - Input (index) the documents
  - Put each query to the system
  - Collect the output

(thanks to Stephen Robertson)
The traditional IR experiment

- Then you need to
  - Evaluate the output, document by document
  - Discover (??) the good documents your system has missed
  - Analyse the results

(thanks to Stephen Robertson)
The traditional IR experiment

- What is a document?
  - traditionally: a package of information structured by an author

- What is a request?
  - a description of a topic of interest
  - more properly, a partial representation of an underlying information need

- What is a system?
  - A device that accepts a request and delivers of identifies documents
  - "device" may be an organisation: involve people(!)

(thanks to Stephen Robertson)
The traditional IR experiment

- Assuming that documents are either relevant or not, the objective is:
  - To retrieve relevant documents
  - Not to retrieve non-relevant documents
The traditional IR experiment

Evaluation measures

- precision = \( \frac{r}{n} \): fraction of retrieved documents that is relevant

- recall = \( \frac{r}{R} \): fraction of relevant documents that is retrieved

\( r \): number of relevant documents retrieved
\( n \): number of documents retrieved
\( R \): number of relevant documents
What about ranked output?

- Report precision for positions in the ranked list
  - 5, 10, 20 document retrieved
- Report precision for some recall levels
  - precision at 0.1, 0.2, etc.
Recall-precision plot
Recall-precision plot
The traditional IR experiment

- Problems with IR system evaluation
  - costly (involves users)
  - which documents did the system miss?
  - hard to repeat in same settings (learning / fatigue effects)
  - we need a complete system(!) we do not in general know how to evaluate components
The TREC paradigm

doing laboratory tests
Benchmark collections

- Consists of three parts:
  - documents (realistic contents and size)
  - requests (textual description of information need; realistic, "real" application)
  - relevance assessments: how useful is the retrieved document?

- How to design?
  - Cranfield → TREC → CLEF, NTCIR, INEX
What is TREC?

- Competition/collaboration between IR research groups world-wide
- Run by the US National Institute of Standards and Technology (NIST)
- TREC provides:
  - common test collections
  - common tasks
  - common measures
  - common evaluation procedures
What is TREC?

- A workshop series that provides the infrastructure for large-scale testing of text retrieval technology
  - realistic test collections
  - uniform, appropriate scoring procedures
  - a forum for the exchange of research ideas and for the discussion of research methodology

(thanks to Ellen Voorhees)
TREC approach

Assessors create topics at NIST

Topics are sent to participants, who return ranking of best 1000 documents per topic

Systems are evaluated using relevance judgments

NIST forms pools of unique documents from all submissions which the assessors judge for relevance

(thanks to Ellen Voorhees)
An example TREC topic

<top>
<num> 405
<title> cosmic events
<desc> What unexpected or unexplained cosmic events or celestial phenomena, such as radiation and supernova outbursts or new comets, have been detected?
<narr> New theories or new interpretations concerning known celestial objects made as a result of new technology are not relevant.
</top>
Creating Relevance Judgments

RUN A

401

RUN B

401

Top 100

Pools

Alphabetized Docnos

401

402

403

(thanks to Ellen Voorhees)

Text REtrieval Conference (TREC)
TREC assumptions about relevance

- Relevance of one element does not affect the relevance of another element.
- Relevance is a binary decision, i.e., a document is either relevant or not.
- A document is relevant if it would help in writing an article about the subject.
TREC assumptions about systems

- A system is a programme
  - the user is outside the system
- A system is an input-output device
  - query in, documents out
  - although... most real searches involve interaction
How about the quality of a test collection?

- Two concerns:
  - **Consistency** of the judgments: *do the results of the experiments critically depend on the particular choices of human judges?*
  - **Completeness** of the judgments: *do the results critically depend on the pool construction process, i.e. on the systems that participated in TREC?*
Consistency of the judgements

- Experiment: 10 topics assessed twice by two different assessors
- Dutch CLEF collection, overlap: 0.465
- TREC: overlap between: 0.421 and 0.494
  (Overlap = size of intersection of the relevant document sets divided by the size of the union of the relevant document sets.)
- (Overall agreement 93.4 %)
Completeness of judgments

- Can we use the collection for future experiments?
- What if my run is not judged?
- Experiment: recompute for each official run the average precision as if it was not in the pool, i.e. ignoring the relevant documents uniquely found by that run
### Completeness of the judgments:
What if my run is not judged?

<table>
<thead>
<tr>
<th>run name</th>
<th>unjudged / judged avg. prec.</th>
<th>difference</th>
<th>unique rel.</th>
</tr>
</thead>
<tbody>
<tr>
<td>ut1</td>
<td>0.4222 0.4230</td>
<td>0.0008 0.2 %</td>
<td>55</td>
</tr>
<tr>
<td>aplmonla</td>
<td>0.3943 0.4002</td>
<td>0.0059 1.5 %</td>
<td>29</td>
</tr>
<tr>
<td>tnonn3</td>
<td>0.3914 0.3917</td>
<td>0.0003 0.1 %</td>
<td>2</td>
</tr>
<tr>
<td>humNL01x</td>
<td>0.3825 0.3831</td>
<td>0.0006 0.2 %</td>
<td>5</td>
</tr>
<tr>
<td>tlrnltd</td>
<td>0.3760 0.3775</td>
<td>0.0015 0.4 %</td>
<td>10</td>
</tr>
<tr>
<td>tnoen1</td>
<td>0.3246 0.3336</td>
<td>0.0090 2.8 %</td>
<td>32</td>
</tr>
<tr>
<td>AmsNIM</td>
<td>0.2770 0.2833</td>
<td>0.0063 2.3 %</td>
<td>32</td>
</tr>
<tr>
<td>aplbiennl</td>
<td>0.2692 0.2707</td>
<td>0.0015 0.6 %</td>
<td>7</td>
</tr>
<tr>
<td>oce2</td>
<td>0.2363 0.2405</td>
<td>0.0042 1.8 %</td>
<td>21</td>
</tr>
<tr>
<td>glaenl</td>
<td>0.2113 0.2123</td>
<td>0.0010 0.5 %</td>
<td>8</td>
</tr>
<tr>
<td>oce1</td>
<td>0.2024 0.2066</td>
<td>0.0042 2.1 %</td>
<td>23</td>
</tr>
<tr>
<td>medialab</td>
<td>0.1600 0.1640</td>
<td>0.0040 2.5 %</td>
<td>23</td>
</tr>
<tr>
<td>EidNL2001A</td>
<td>0.1339 0.1352</td>
<td>0.0013 1.0 %</td>
<td>8</td>
</tr>
</tbody>
</table>

**Mean:** 0.0031 1.2 % 20

**Standard deviation:** 0.0027 1.0 % 15
Significance testing

- When is one system better than another?
  - Maybe the average difference can be contributed to chance?
  - Need a reasonable amount of queries (e.g. 50), which should be a random sample of all possible queries for a given task
Significance testing

- Two hypotheses
  - null-hypothesis $H_0$: there is no difference between system $A$ and system $B$
  - alternative hypothesis $H_1$: either system $A$ consistently outperforms system $B$, or system $B$ consistently outperforms system $A$

- Show that, given the evaluation results, $H_0$ is indefensible
Significance testing

- Test statistics should behave differently under $H_0$ than under $H_1$:
  - Paired tests: for each query the performance difference between system A and B consist of a mean difference $\mu$ and some error.
    
    $H_0 : \mu = 0; \; H_1 : \mu \neq 0;$
  
  - Paired t-test: assumes that errors are normally distributed. Under $H_0$ the distribution is Student's t
  
  - Paired sign test: assumes equal probability of positive and negative error. Under $H_0$ the distribution is binomial
Conclusion

- To evaluate your system, use a benchmark collection.
- Choose appropriate evaluation measures
- Base your conclusions on statistical tests
Acknowledgements

Thanks to the following people for making their slides available

- Stephen Robertson (Microsoft Research)
- Ellen Voorhees (NIST)
Some background reading

