Ranking Clusters for Web Search

Gianluca Demartini
Paul–Alexandru Chirita
Ingo Brunkhorst
Wolfgang Nejdl

L3S Info Lunch
Hannover, 08 November 2006

Outline

- Introduction
- Rankings Algorithms considered
- Experimental Setup
- Results
- Conclusions

Introduction (1)

- Search the Web: results are presented sorted using a score value
- Users should be able to browse the results efficiently
- An interface that clusters documents performs better
- Common task in Clustering Search Engines (SE): ordering the results of the classification
- An efficient ordering of the clusters will be benefic for the user

Introduction (2)

- We analyze a set of ten different metrics for ordering clusters of search engine result:
  - Ranking by SE Scores
  - Ranking by Query to Cluster Similarity
  - Ranking by Intra Cluster Similarity
  - Measures independent of the documents within the cluster

- Two different clustering algorithms: performances of the cluster rankings is not dependent of the clustering algorithms used

Related Work (1)

- SE already employ such an output structuring: Vivisimo, iBoogie, Mooter, Grokker, etc.
- Many Techniques to cluster web search results: flat manner, or in a hierarchical way
- Clustering useful for clarifying a vague query, by showing the dominant themes

Related Work (2)

- How to display search results to the users: they find answers faster using a categorized organization
- Faceted search: an Alphabetical order is commonly utilized
- Text Classifiers: SVM better than Bayesian for Text Classification

Outline

- Introduction
- Rankings Algorithms considered
- Experimental Setup
- Results
- Conclusions

Cluster Ranking Algorithms

- 10 different ranking algorithms considered:
  - Ranking by search engine scores (4)
  - Ranking by Query to Cluster Similarity (1)
  - Ranking by Intra Cluster Similarity (2)
  - Measures independent of the documents within the cluster (3)

Ranking by search engine scores (1)

- PageRank computation: $PRv = x^{-2.1}$ *page* at position x

- Average PageRank $\mathrm{AvgPR}(C) = \frac{1}{n}\sum_{p=1}^{n} \mathrm{PRv}(p),\ \forall \text{ page } p \in C$

- Total PageRank $\mathrm{SumPR}(C) = \sum_{p=1}^{n} \mathrm{PRv}(p),\ \forall \text{ page } p \in C$

Ranking by search engine scores (2)

- Average Rank

$$\text{AvgRank}(C) = \frac{1}{n}\sum_{p=1}^{n} \text{Rank}(p),\ \forall \text{ page } p \in C$$

- Minimum Rank

$$\text{MinRank}(C) = \min_{p} \text{ Rank(p)},\ \forall \text{ page } p \in C$$

Ranking by Query to Cluster Similarity

- Normalized Logarithmic Likelihood Ratio

$$\mathrm{NLLR}(q,p)=\sum_{t\in q}P(t|p)*\log\frac{(1-\lambda)\cdot P(t|p)+\lambda\cdot P(t|C)}{\lambda\cdot P(t|C)}$$

- Average Query/Page similarity

$$\mathrm{AvgSimilarity}(C)=\frac{1}{n}\sum_{p=1}^{n}\mathrm{NLLR(Q,p)},\ \forall\ \mathrm{page}\ p\in C$$

Ranking by Intra Cluster Similarity

- Similarity between pages and categories (title + description)
  - values returned by the classifiers
  - probability that a document belongs to some category
  - strength with which every result belongs to its assigned category

- **Average Intra Cluster Similarity**. (AvgValue)
  - over all the pages that belong to a category
  - to the top of the list, clusters where the results are most relevant to their category

- **Maximum Intra Cluster Similarity**. (MaxValue)
  - the focus is on the best match-ing document of each cluster only
  - the results the user views first are those that have been best classified

Other Metrics

- Metrics which seem to be used by current commercial web SE and a baseline
- Order by **Size**
  - using the number of docs belonging to the category
  - used by most of the Clustering SE (e.g. Vivisimo)
- **Alphabetical** Order
  - used in Faceted Search (e.g. Flamenco)
- **Random** Order
  - to compare the other metrics

Outline

- Introduction
- Basic Concepts
- Rankings Algorithms considered
- Experimental Setup
- Results
- Conclusions

Experimental Setup (1)

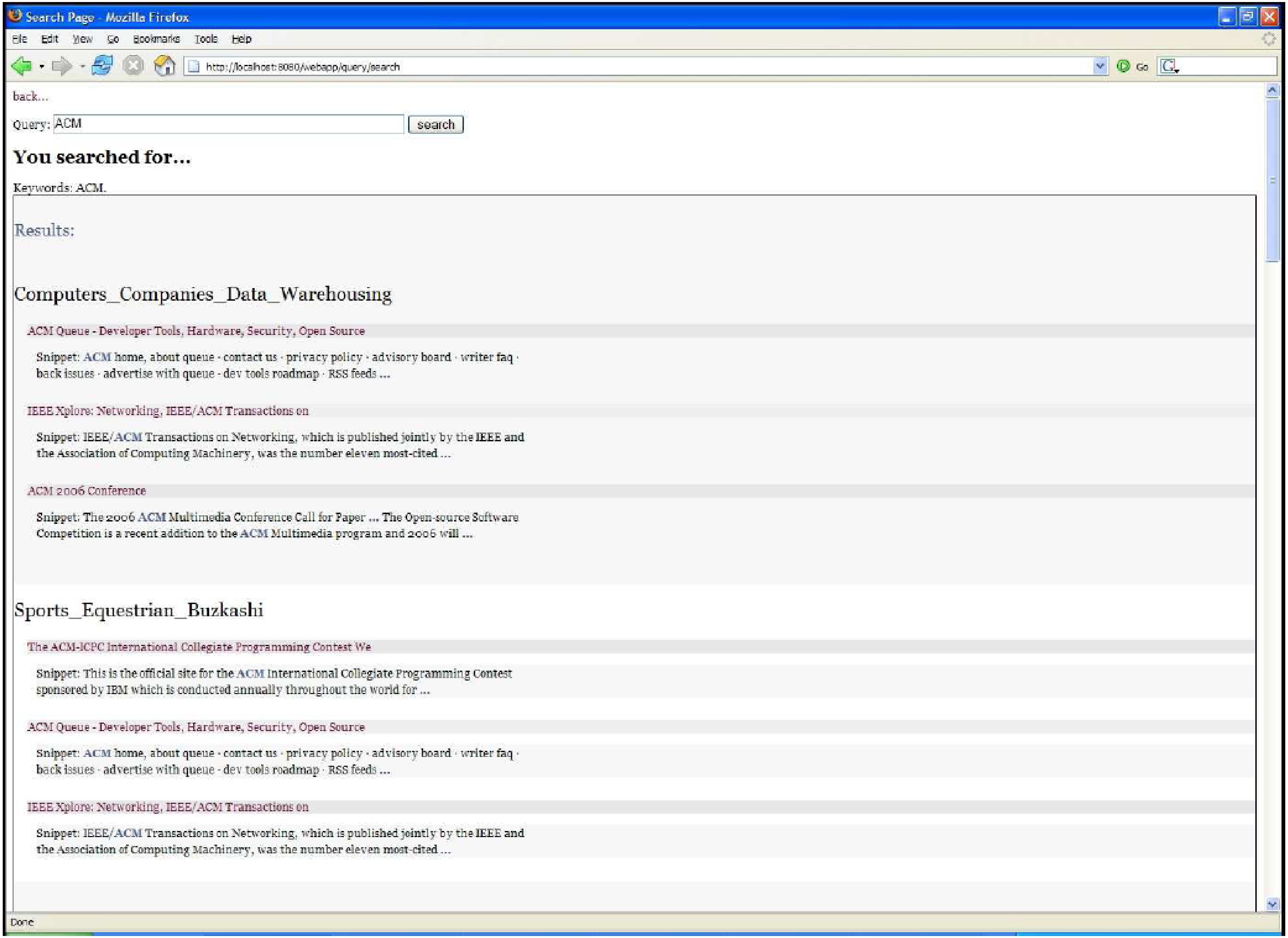
- 20 algorithms (10 ranks, 2 classifiers), 20 people
- Supporting Vector Machines (SVM) and Bayes as Text Classifiers
  - the performance of the ranking algorithms considered does not depend on the clustering algorithm used
- ODP categories (top 3 levels)
- 50 000 most frequent terms in ODP titles and descriptions of web pages
- 5 894 English categories

Experimental Setup (2)

- Each user evaluated each (algorithm,classifier) once:
  - task: select the first relevant result
  - no information about which algorithm was being used
  - subject began the evaluation from different algorithms
  - the order of results within a category is the one of Google
- We **measure** the **time** spent for search the relevant result and the **position** of the results
- Each user 20 query:
  - 12 from Topic Distillation Task of the Web Track 2003
  - 8 from TREC Web Track 2004 (4 of them ambiguous)
  - one extra query at the beginning for getting familiarized

Experimental Setup (3)

- Classification:
  - retrieved titles and snippets of the top 50 results from Google
  - allowed each result to belong to maximum three categories (the ones with the best similarity values)
  - showed to the user only the top 75 results after ranking the clusters to put emphasis on the performances of the ranking
  - all the results were cached to ensure that results from different participants were comparable

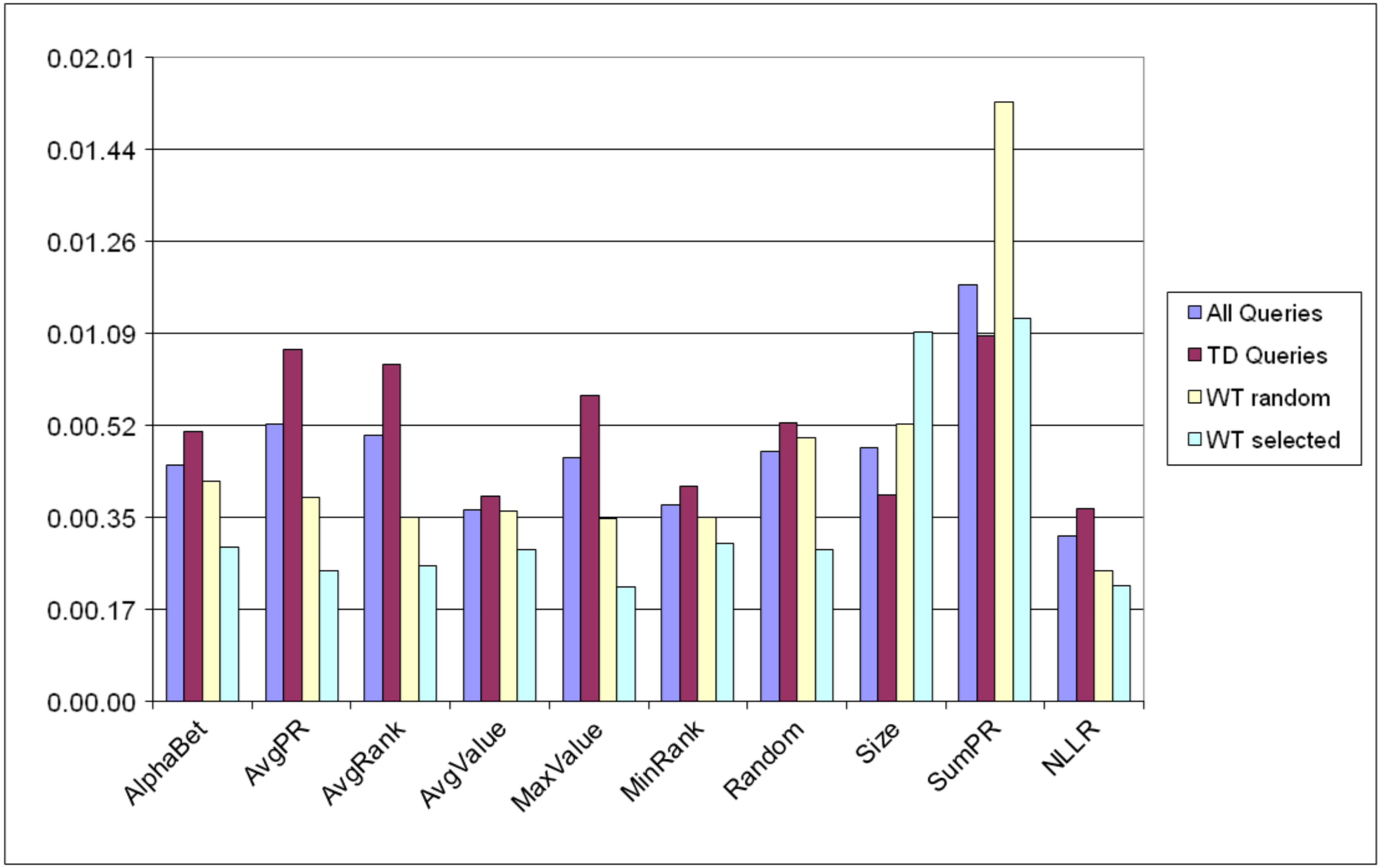

Search Page - Mozilla Firefox
File Edit View Go Bookmarks Tools Help
http://localhost:8080/webapp/query/search
Go
back...
Query: ACM
search
You searched for...
Keywords: ACM.
Results:
Computers_Companies_Data_Warehousing
ACM Queue - Developer Tools, Hardware, Security, Open Source
Snippet: ACM home, about queue · contact us · privacy policy · advisory board · writer faq · back issues · advertise with queue · dev tools roadmap · RSS feeds ...
IEEE Xplore: Networking, IEEE/ACM Transactions on
Snippet: IEEE/ACM Transactions on Networking, which is published jointly by the IEEE and the Association of Computing Machinery, was the number eleven most-cited ...
ACM 2006 Conference
Snippet: The 2006 ACM Multimedia Conference Call for Paper ... The Open-source Software Competition is a recent addition to the ACM Multimedia program and 2006 will ...
Sports_Equestrian_Buzkashi
The ACM-ICPC International Collegiate Programming Contest We
Snippet: This is the official site for the ACM International Collegiate Programming Contest sponsored by IBM which is conducted annually throughout the world for ...
ACM Queue - Developer Tools, Hardware, Security, Open Source
Snippet: ACM home, about queue · contact us · privacy policy · advisory board · writer faq · back issues · advertise with queue · dev tools roadmap · RSS feeds ...
IEEE Xplore: Networking, IEEE/ACM Transactions on
Snippet: IEEE/ACM Transactions on Networking, which is published jointly by the IEEE and the Association of Computing Machinery, was the number eleven most-cited ...
Done

Outline

- Introduction
- Basic Concepts
- Rankings Algorithms considered
- Experimental Setup
- Results
- Conclusions

Experimental Results

- Time to find the relevant result

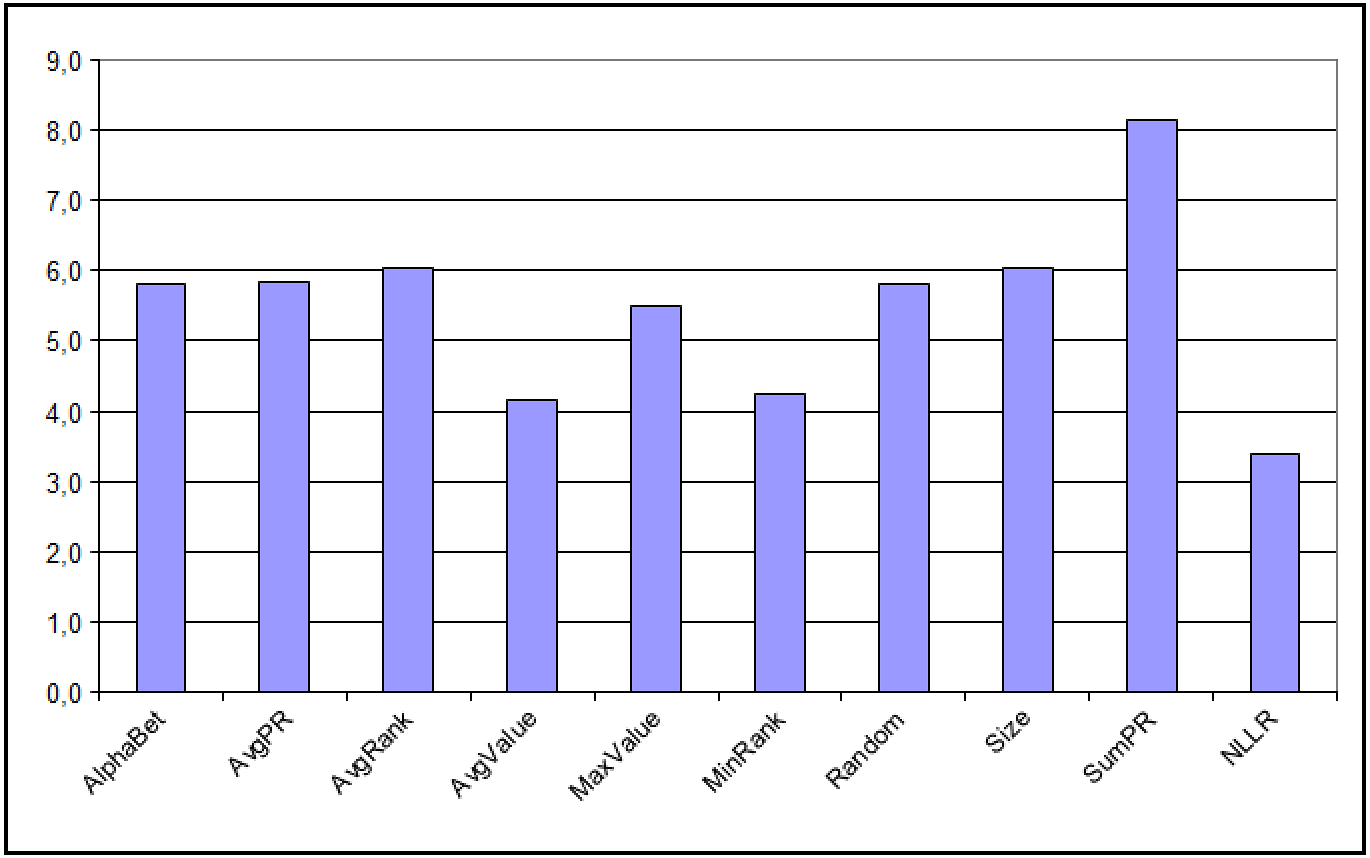


Experimental Results

- Time to find the relevant result
  - **NLLR** allowed the user to find relevant results in **the fastest** way, with an average of 31s
  - performances of **Alphabetical** and the **Size** based rankings is rather **average**
  - Topic Distillation ones have been **the most difficult**: they have a task associated
  - Web Track ambiguous ones were **the easiest**: no specific search task was associated, and thus the first relevant result was easier to find
  - experiment is **statistically significant** at a 99% confidence level.

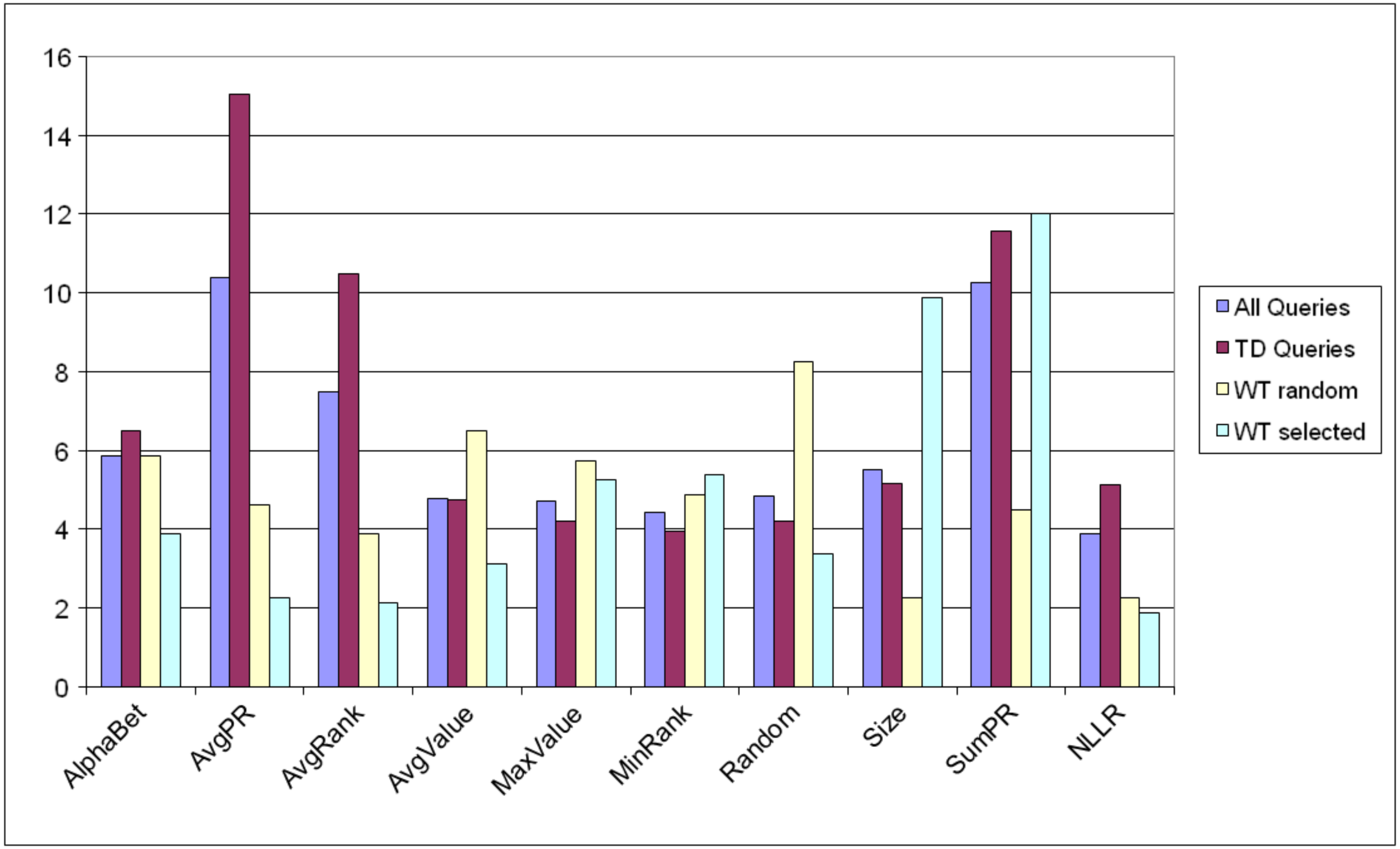
Experimental Results

- Average of the position of the algorithm for each user



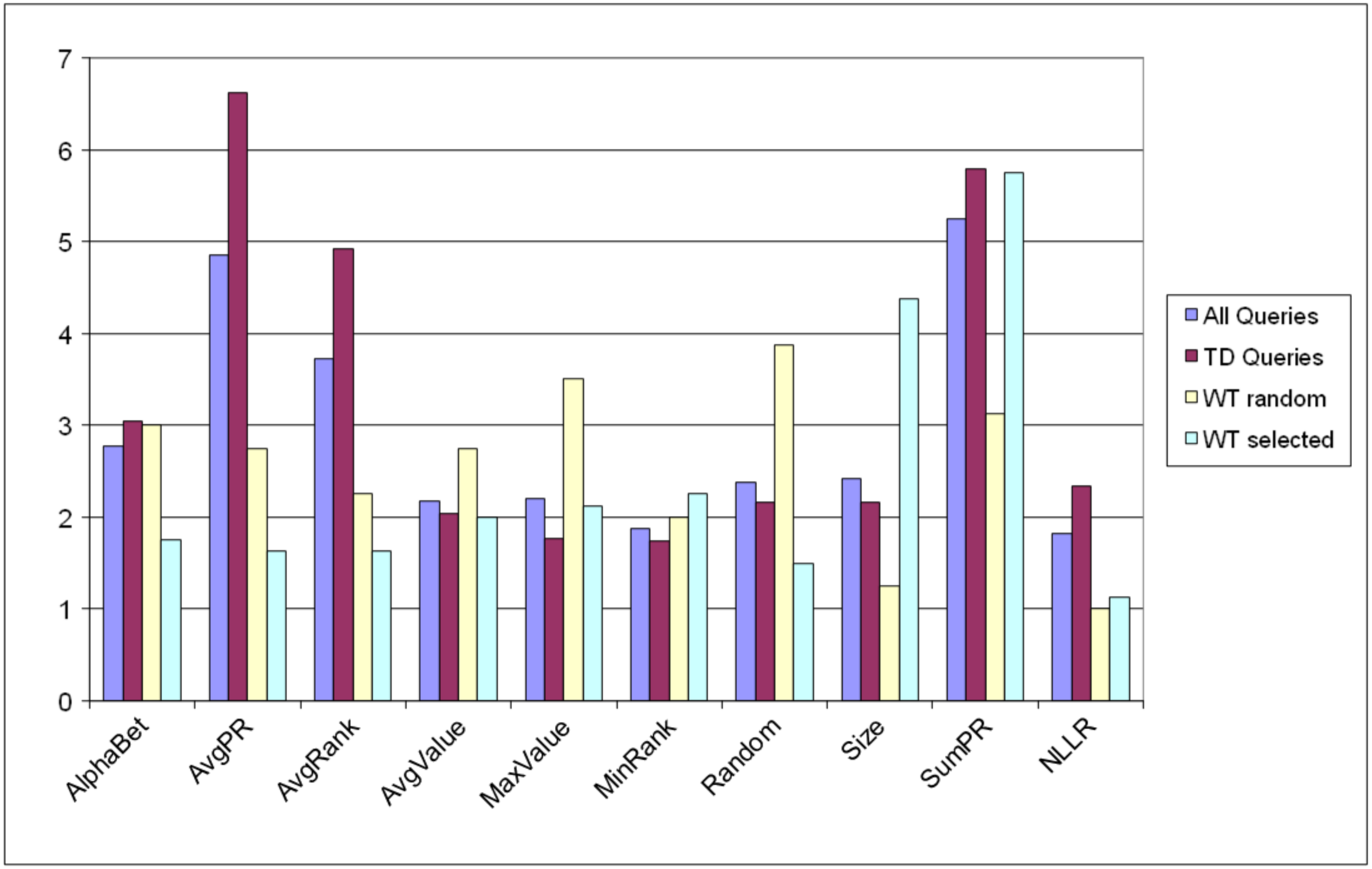
Experimental Results

- Average Rank of the Result



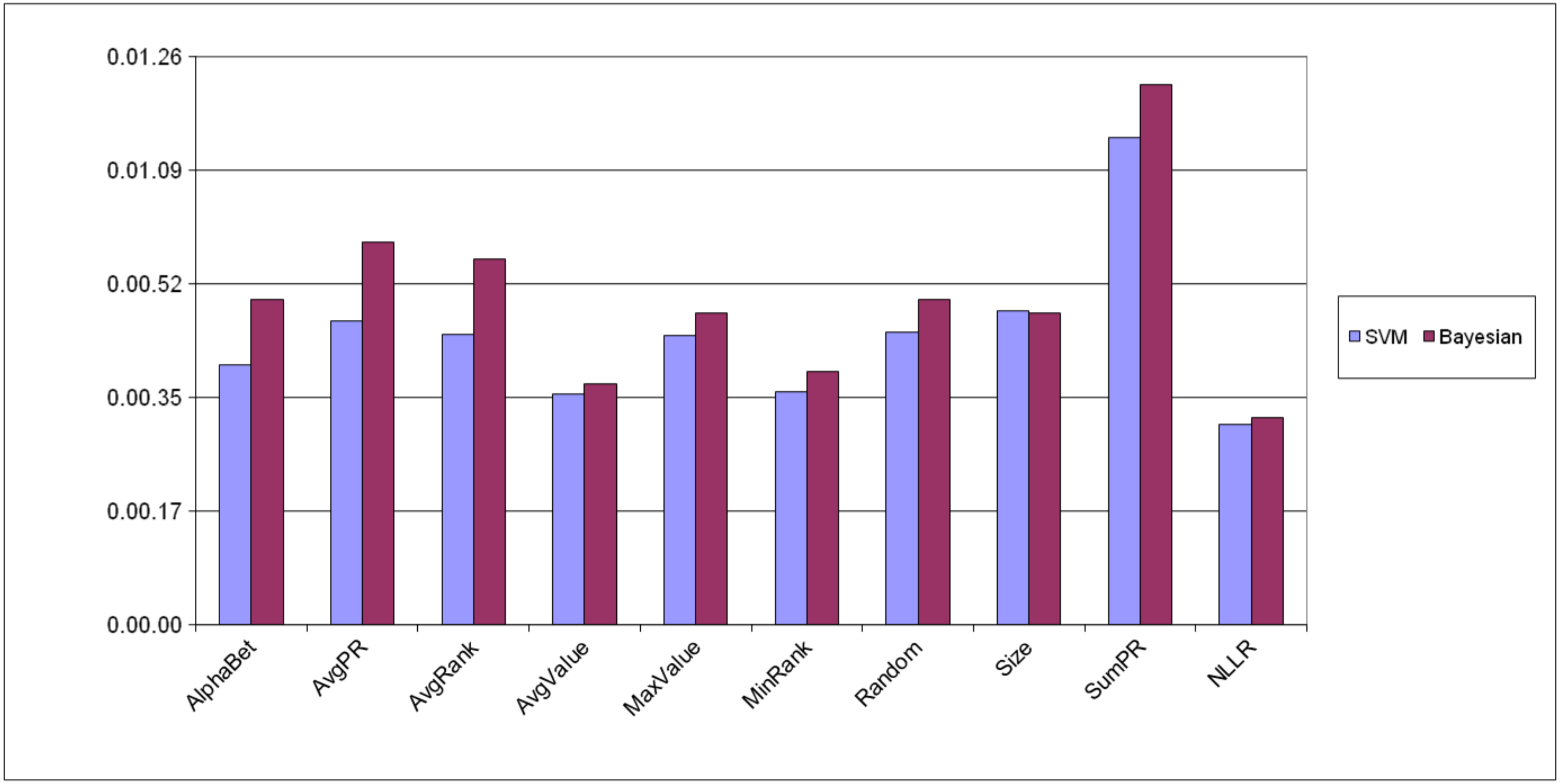
Experimental Results

- Average Rank of the Cluster



Experimental Results

- The results are slightly better when using SVM



Conclusions & Future Work

- Similarity between the user query and the documents seems to be the best approach to order search result clusters

- Alphabetical and Size Ranking are not so good

- We want to test other algorithms
  - click-thorought data
  - clustering algorithms which produce results more apart from each other

Thanks for your attention!

Q&A