Keeping ADS relevant

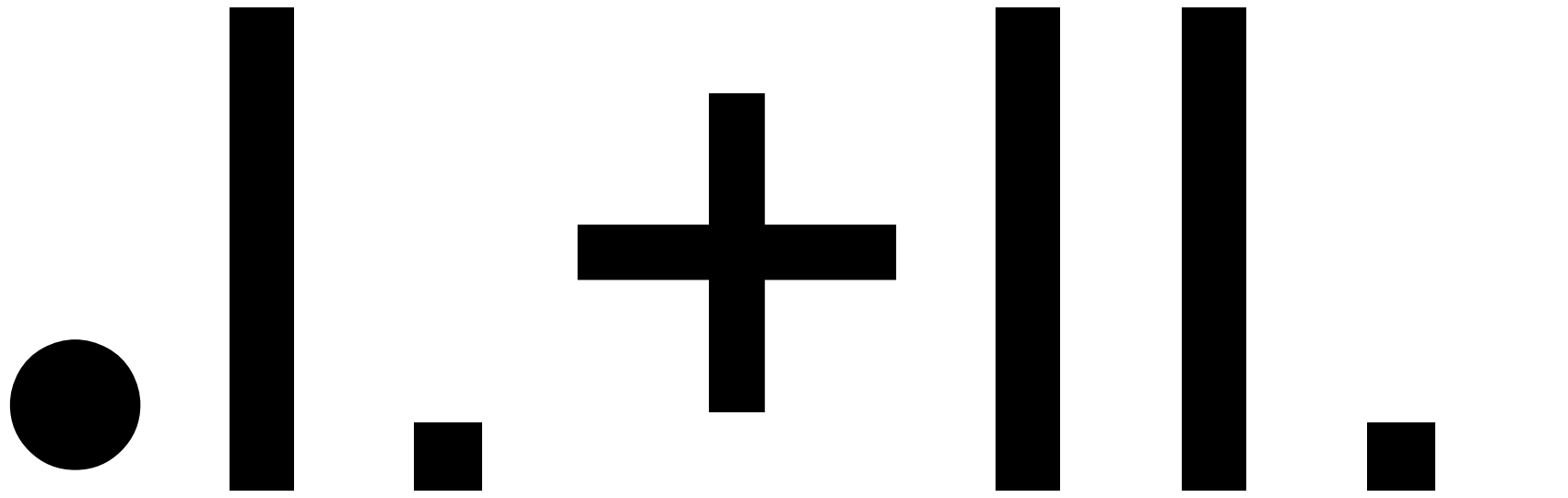
Problem Statement

type	**frequency**	**difficulty**
person	80%	10%
copy&paste	10%	10%
topic	10%	80%

3 pillars of the search

- How relevant (Results Ranking)
- How knowledgeable (about THE User)
- How fast (Search Speed and Capacity)

Good enough will have to suffice



- Search Relevancy
- Knowing the user

Relevance plan

- Port Classic ranking
- Get infrastructure for intelligent learning
  - Collecting data (about users, queries, results)
  - Evaluating/measuring impact of used variables
- Update, rinse, repeat

Baseline Relevance

- Classic ranking
  - ½ of score contributed by the match between the query and the paper
  - ½ of score contributed by quality of the paper
    - log(1 + #citations + normalized_reads)
  - Works well for metadata queries
  - Implementation
    - Slightly different scoring model (custom component)
      - Normalization is applied to the final score
      - Not to the matching query components
    - It can be done (actually, we have had this functionality for a long time), but need precision for two questions
      - What's the impact on search/load?
      - Can we avoid making customizations to SOLR?

Example query: LSST

- (((abstract:acr::lsst abstract:syn::acr::lsst abstract:syn::large synoptic survey telescope))^1.3
- ((author:lsst, author:lsst,*))^2.0
- ((title:acr::lsst title:syn::acr::lsst title:syn::large synoptic survey telescope))^1.5

Amusing (at least to me) query

THE → ((abstract:acr::the)^1.3 | ((author:the, author:thè, author:thé, author:thé,;thè, author:the,* author:thè,* author:thè, author:thè,* author:thé,* author:thee, author:thee,* author:thé,;thè, * author:thé, ; author:thé, ; * author:thee,;thè, author:thee,;thè, * author:thee, ; author:thee, ; * author:the,;the, author:the,;the, * author:the, ; author:the, ; *))^2.0 | bibstem:the | ((first_author:the, first_author:thè, first_author:thé, first_author:thé,;thè, first_author:the,* first_author:thè,* first_author:thè, first_author:thè,* first_author:thé,* first_author:thee, first_author:thee,* first_author:thé,;thè, * first_author:thé, ; first_author:thé, ; * first_author:thee,;thè, first_author:thee,;thè, * first_author:thee, ; first_author:thee, ; * first_author:the,;the, first_author:the,;the, * first_author:the, ; first_author:the, ; *))^5.0 | identifier:the | (title:acr::the)^1.5 | (year:the)^2.0)",

Baseline Relevance

- For fulltext search
  - Either a combination of constant scores (sort of mimicking Classic behaviour)
  - Or combination of damping boost factors across fields (when searching across indexes), i.e. unfielded search
    - first_author^15
    - author^10
    - title^8 …..
- Multitude of search features already in place
  - Boosting, unfielded search, synonyms…
  - Too many to list (over hundred, but that's OK, they are all well tested)

Learning to Rank

- Search features
  - Query specific
  - Document specific
  - User specific
- Most promising features are “external” to the document/query
- But impact of each individual feature is difficult to measure
  - Need to collect data
  - Turn data into signals

Scoring Simulateur
Dashboard
Experiment Setup
Returned Documents
Selected Relevant Papers
Experiment Results

☐	0.1836734693877551	0.8999999999999999	0.9009999999999999	false	true

Number of combinations explored: 9360, Time elapsed: 574.750619 s., Progress: 1

New Score ↓	Lucene Score	Relevant	Title	Authors	Publication
13.440000000000001	12	☑	Creation and Use of Citations in the ADS	Accomazzi, A.; Eichhorn, G.; Kurtz, M. J.; Grant, C. S.; Henneken, E.; Demleitner, M.; Thompson, D.; Bohlen, E.; Murray, S. S.	
10.2	12	☑	The Future of Technical Libraries	Kurtz, M. J.; Eichhorn, G.; Accomazzi, A.; Grant, C.; Henneken, E.; Thompson, D.; Bohlen, E.; Murray, S. S.	
10.2	10	☐	E-prints and journal articles in astronomy: a productive co-existence	Henneken, Edwin A.; Kurtz, Michael J.; Eichhorn, Guenther; Accomazzi, Alberto; Grant, Carolyn S.; Thompson, Donna; Bohlen, Elizabeth; Murray, Stephen S.; Ginsparg, Paul; Warner, Simeon	
				Kurtz, Michael J.; Henneken, E. A.;	

Learning to Rank

- Simulateur (adsabs.harvard.edu/scorer)
  - Platform for simulating query response
  - Grid search for optimal set of parameters
  - Types of data
    - Expert judgment
    - *Classic results*
    - *User clicks*

Collecting signals

- We are going to collect more data
  - About users
  - About their actions
- Yet the data must be easily accessible
  - Temporal (time series database)
- Actionable
  - **Eventually** we'll plug this data into the search algorithm (online)

ads

Feedback | ORCID | About | Account

QUICK FIELD: Author | First Author | Abstract | Year | Fulltext | All Search Terms

Start New Search

abs:"extractive summarization"

Your search returned **112** results

Score | Export | Explore

AUTHORS
- Lapata, M 5
- Wei, F 4
- Zhou, M 4
- Liu, F 3
- Wang, L 3

more

COLLECTIONS
- general 85
- physics 29
- astronomy 3

REFEREED
- non-refereed 96
- refereed 16

KEYWORDS

PUBLICATIONS

BIB GROUPS

SIMBAD OBJECTS

NED OBJECTS

DATA

VIZIER TABLES

Hide highlights | Show abstracts | Hide Sidebars | Go To Bottom

1 2017arXiv170404530N 2017/04 cited: 4
Neural Extractive Summarization with Side Information
Narayan, Shashi; Papasarantopoulos, Nikos; Cohen, Shay B. *and 1 more*

> *Neural Extractive Summarization with Side Information*
> *Most extractive summarization methods focus on the main body of the document from which sentences*

2 2018arXiv180909672D 2018/09
BanditSum: Extractive Summarization as a Contextual Bandit
Dong, Yue; Shen, Yikang; Crawford, Eric *and 2 more*

> *BanditSum: Extractive Summarization as a Contextual Bandit*
> *-document extractive summarization without heuristically-generated extractive labels. We call our approach BanditSum*

3 2018arXiv181012085A 2018/10
Extractive Summarization of EHR Discharge Notes
Alsentzer, Emily; Kim, Anne

> *Extractive Summarization of EHR Discharge Notes*
> *a dataset for evaluation of extractive summarization methods.*

4 2018arXiv180510392A 2018/05
Reinforced Extractive Summarization with Question-Focused Rewards
Arumae, Kristjan; Liu, Fei

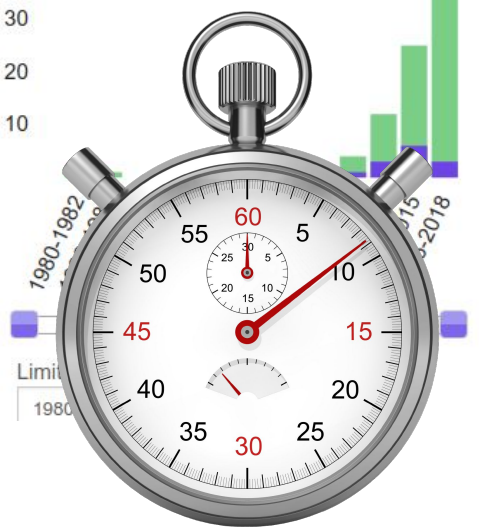
0 selected

Add papers to library

Years | Citations | Reads

refereed non refereed

60 50 40 30 20 10



VIEW

Abstract

Citations (4)

References (1)

Co-Reads

Volume Content

Graphics

Metrics

Export

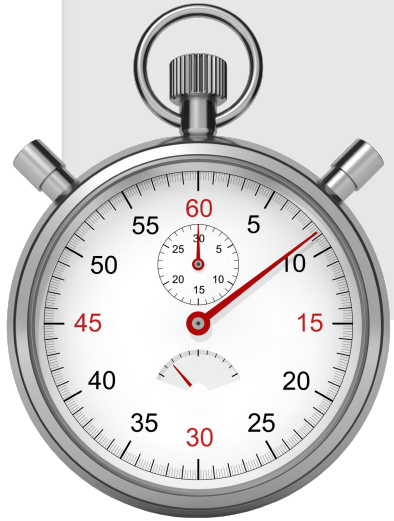
Neural Extractive Summarization with Side Information

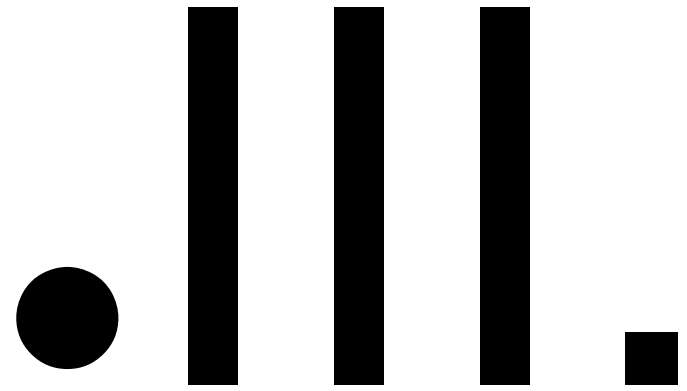
Narayan, Shashi; Papasarantopoulos, Nikos; Cohen, Shay B.; Lapata, Mirella

Most extractive summarization methods focus on the main body of the document from which sentences need to be extracted. However, the gist of the document may lie in side information, such as the title and image captions which are often available for newswire articles. We propose to explore side information in the context of single-document extractive summarization. We develop a framework for single-document summarization composed of a hierarchical document encoder and an attention-based extractor with attention over side information. We evaluate our model on a large scale news dataset. We show that extractive summarization with side information consistently outperforms its counterpart that does not use any side information, in terms of both informativeness and fluency.

Pub Date: April 2017

Bibcode 2017arXiv170404530N

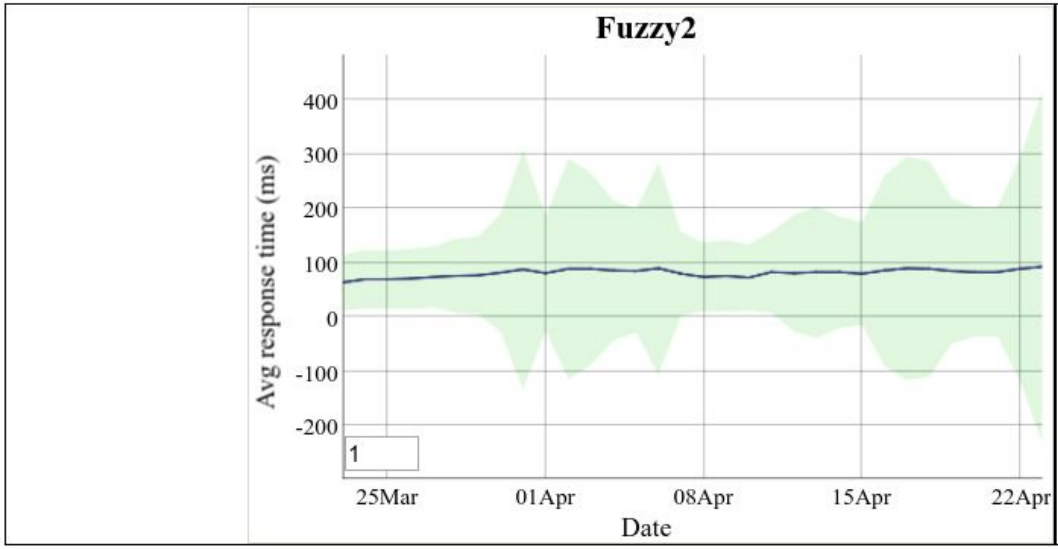


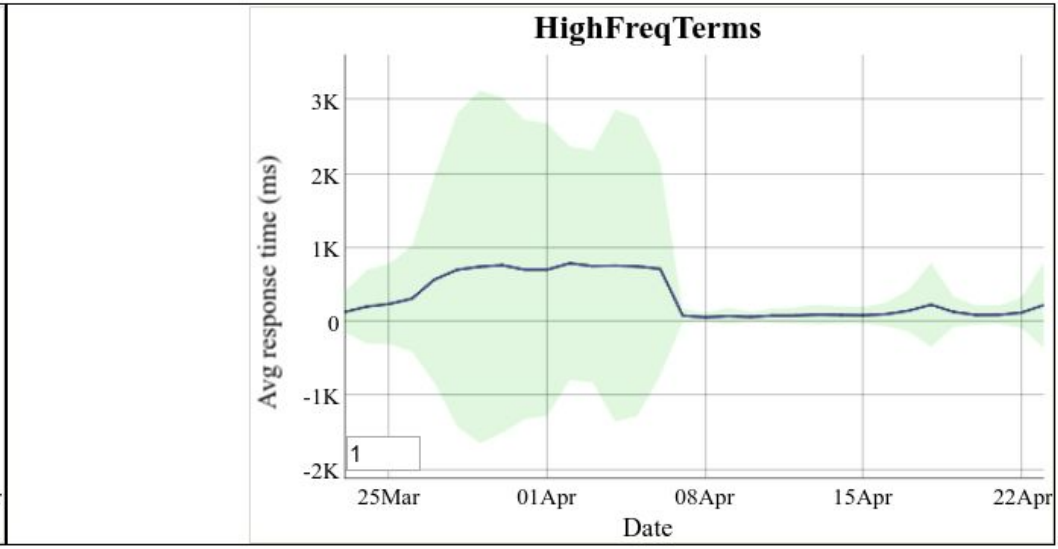


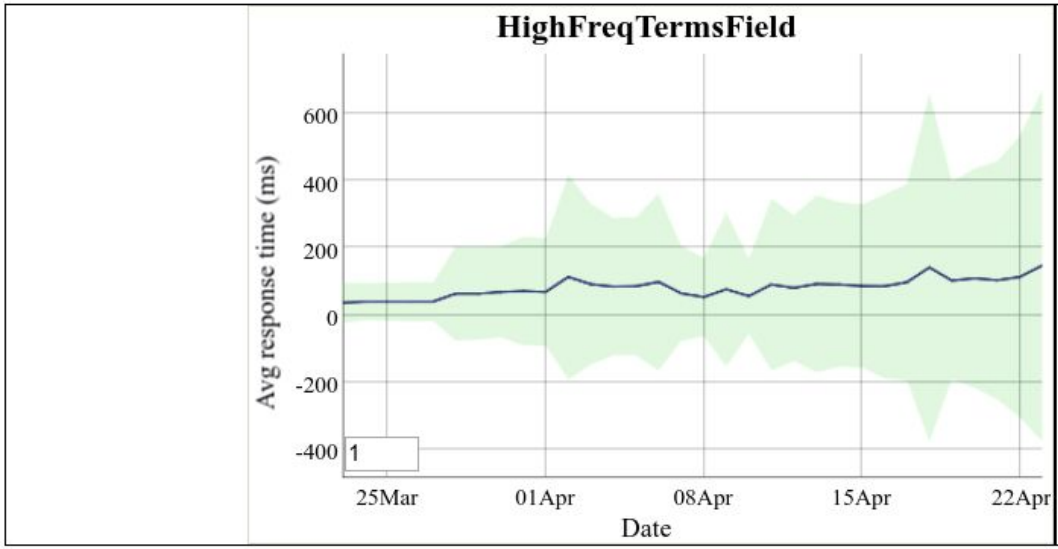
- Search Speed

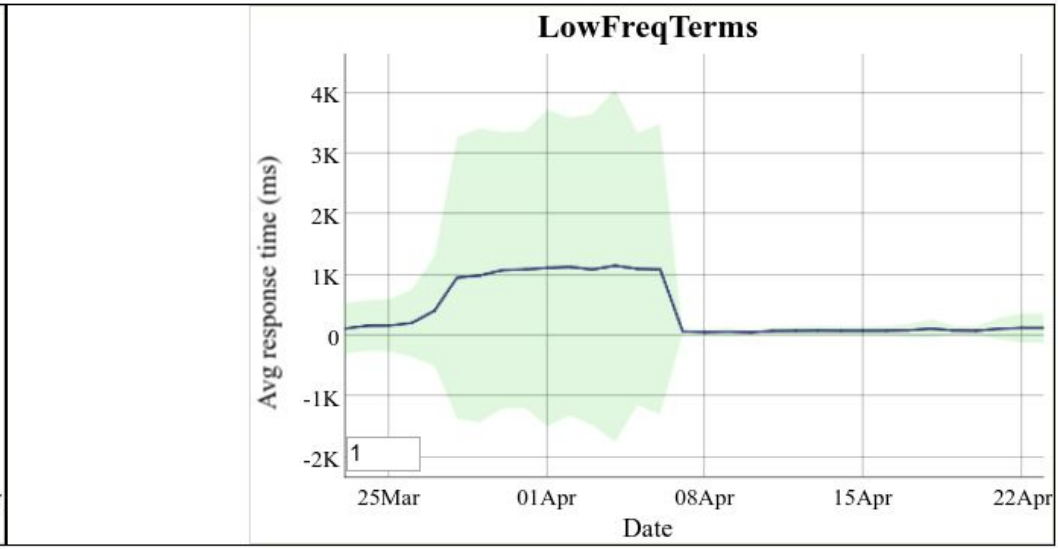
Search Speed

- Index size decreased by 40%
- Put in place detailed performance measurements
  - But we are not yet using them on a regular basis
- Optimized citation cache creation
  - Caused big problems in production
  - New code ready for deployment/testing


Fuzzy2
Avg response time (ms)
400
300
200
100
0
-100
-200
1
25Mar
01Apr
08Apr
15Apr
22Apr
Date


HighFreqTerms
Avg response time (ms)
3K
2K
1K
0
-1K
-2K
1
25Mar
01Apr
08Apr
15Apr
22Apr
Date


HighFreqTermsField
Avg response time (ms)
600
400
200
0
-200
-400
1
25Mar
01Apr
08Apr
15Apr
22Apr
Date


LowFreqTerms
Avg response time (ms)
4K
3K
2K
1K
0
-1K
-2K
1
25Mar
01Apr
08Apr
15Apr
22Apr
Date

Speed

- Some work still remains to be done
  - Ascertain how many nodes/machines we need to run
  - What budget
  - Effective scaling up/down
- In the end we'll have to do what is needed
  - To make user experience fast
  - Even if that might be ugly (separate small/big instances, etc.)

Search Capacity

- Current model
  - Slave/master
    - Good enough for now
    - If index continues to grow lineary
  - Distributed (cloud mode)
    - Necessary if ADS were to index more
    - Second order operations are however a big problem
      - How to do the computation in a distributed fashion

Final notes

- The goals are big
  - We are deliberately aiming high (or one might add: setting ourselves for a failure)
  - But if half is accomplished, the ADS will be in a very good shape for the future
    - Competitive against any similar project
    - But the goal is to be the best, n'est pas?