Scientific Table Search Using Keyword Queries

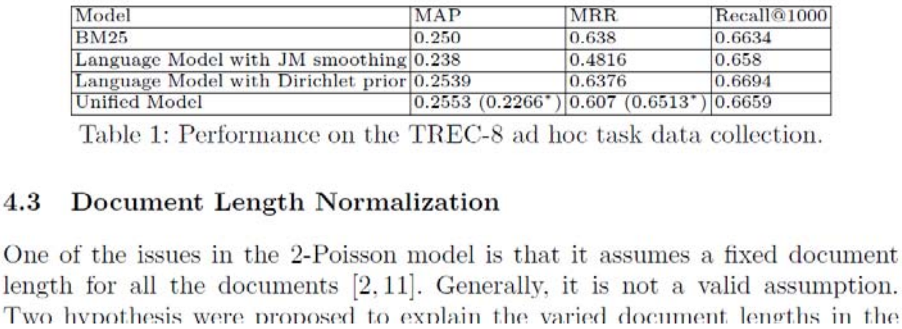
Kyle Yingkai Gao[*] and Jamie Callan
Language Technologies Institute
School of Computer Science
Carnegie Mellon University

[*] Now at Benevolent.AI

Table Retrieval

Task: Table retrieval from scientific publications

- E.g., find MAP values for TREC-8 adhoc corpus



Model	MAP	MRR	Recall@1000
BM25	0.250	0.638	0.6634
Language Model with JM smoothing	0.238	0.4816	0.658
Language Model with Dirichlet prior	0.2539	0.6376	0.6694
Unified Model	0.2553 (0.2266*)	0.607 (0.6513*)	0.6659

Table 1: Performance on the TREC-8 ad hoc task data collection.

4.3 Document Length Normalization

One of the issues in the 2-Poisson model is that it assumes a fixed document length for all the documents [2, 11]. Generally, it is not a valid assumption. Two hypothesis were proposed to explain the varied document lengths in the

(Gorla, Robertson, and Wang, 2011)

Represent Each Table as an XML Document

Many tables aren't described well by their contents

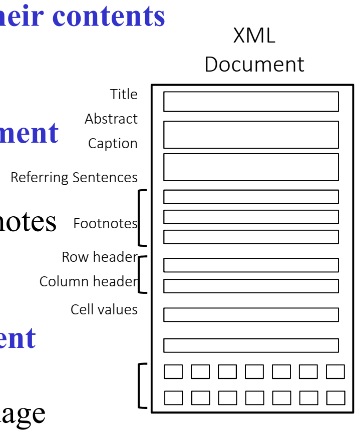
- Meaning is derived from context

Represent each table by an XML document

- Paper title, paper abstract
- Table caption, referring sentences, footnotes
- Row header, column header, cell values

Now it is a standard structured document (XML) retrieval problem

- I.e., it's all about mapping natural language queries to good structured queries



arXiv:1707.03423

Queries

Unstructured queries are mapped to structured queries

- **Query:** gravitational forces in newtonian gravity versus bimetric gravity
- **Entities:** gravitational_force, newtonian_gravity, versus
  - Recognize entities with TagMe
- **Noun phrases:** 'gravitational force', 'newtonian gravity', 'bimetric gravity'
  - Recognize noun phrases with MontyLingua
- **Quantities:** *Force*, *acceleration*
  - Use QUDT to get quantities for query entities & noun phrases

arXiv:1707.03423

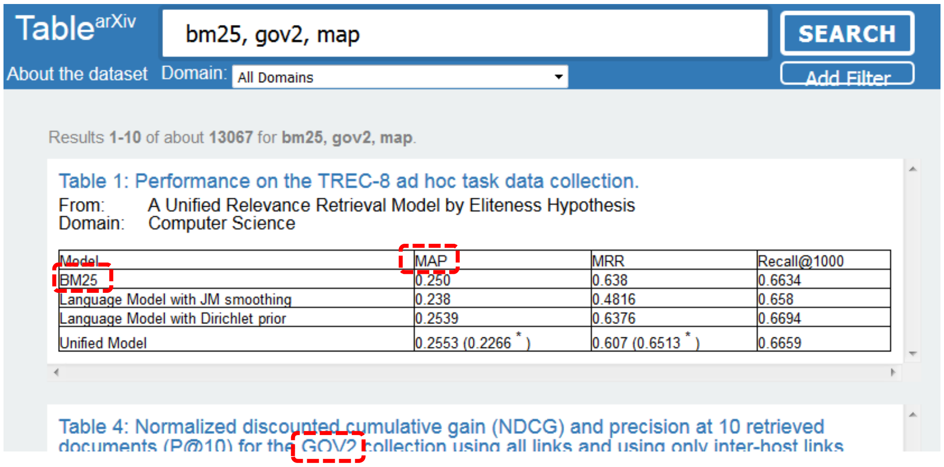
Queries

Unstructured queries are mapped to structured queries

- #wand ((1–α–β) $\text{query}_{\text{terms}}$ α $\text{query}_{\text{concepts}}$ β $\text{query}_{\text{quantities}}$)
  - Multi-field subqueries for terms
  - Multi-field SDM subqueries for concepts
  - Multi-field subqueries for quantities
  - Weights set by parameter sweeps
- A complex query template, but standard IR concepts
  - See the paper for details

arXiv:1707.03423

The Table$^{\text{arXiv}}$ System



Results **1-10** of about **13067** for **bm25, gov2, map**.

Table 1: Performance on the TREC-8 ad hoc task data collection.

From: A Unified Relevance Retrieval Model by Eliteness Hypothesis
Domain: Computer Science

Model	MAP	MRR	Recall@1000
BM25	0.250	0.638	0.6634
Language Model with JM smoothing	0.238	0.4816	0.658
Language Model with Dirichlet prior	0.2539	0.6376	0.6694
Unified Model	0.2553 (0.2266 *)	0.607 (0.6513 *)	0.6659

Table 4: Normalized discounted cumulative gain (NDCG) and precision at 10 retrieved documents (P@10) for the GOV2 collection using all links and using only inter-host links

TableArXiv Dataset

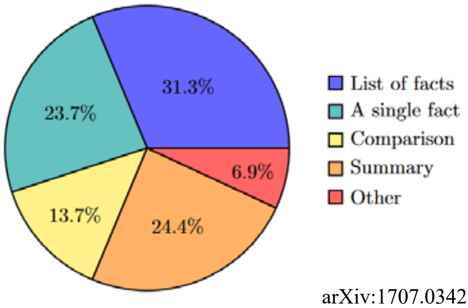
Extract papers with tables from the Physics part of arXiv.org

- 341,573 papers

Hire 8 students with Physics skills to create TREC-like queries

- Use multiple systems to retrieve tables
- Assess manually

Dataset available from my website

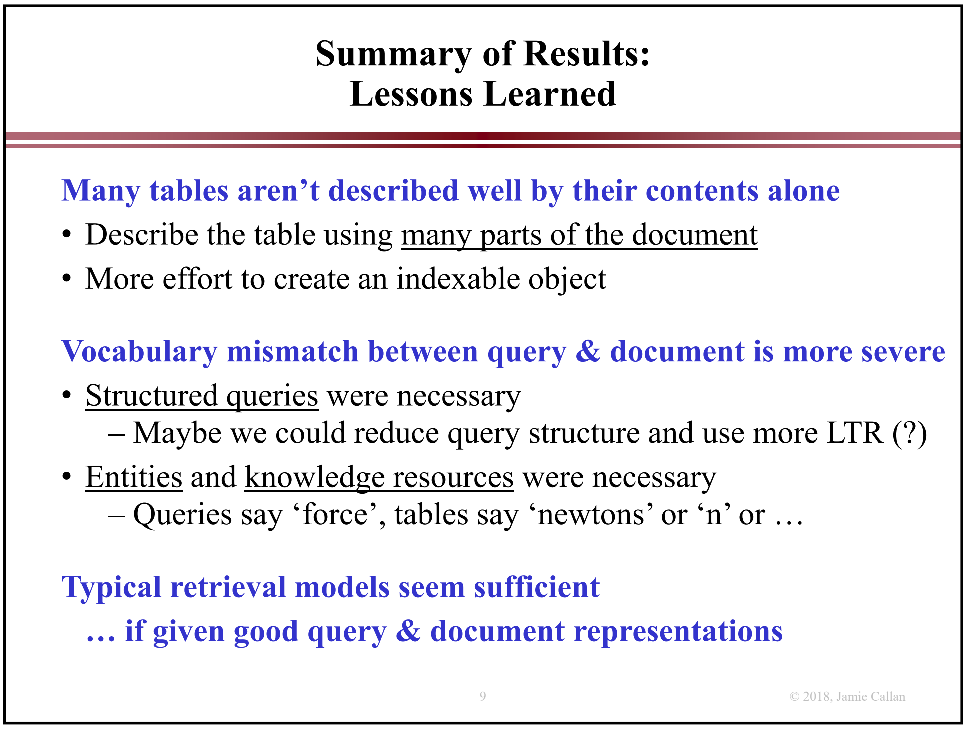


arXiv:1707.03423

Summary of Results

Table^{arXiv} is superior to all baselines

- Of course, otherwise I wouldn't be here ☺
- Results suppressed due to time – see the paper



Summary of Results: Lessons Learned

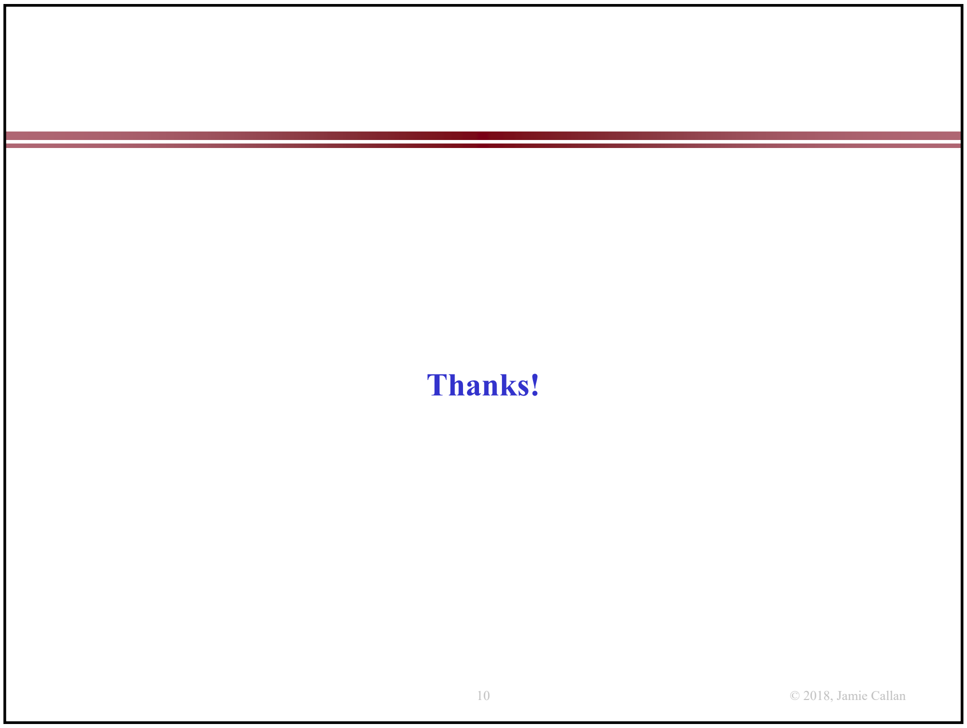
Many tables aren't described well by their contents alone

- Describe the table using <u>many parts of the document</u>
- More effort to create an indexable object

Vocabulary mismatch between query & document is more severe

- <u>Structured queries</u> were necessary
  - Maybe we could reduce query structure and use more LTR (?)
- <u>Entities</u> and <u>knowledge resources</u> were necessary
  - Queries say 'force', tables say 'newtons' or 'n' or …

Typical retrieval models seem sufficient
… if given good query & document representations



Thanks!