Table Search

SIGIR 2019 tutorial - Part IV

Shuo Zhang and **Krisztian Balog**

University of Stavanger

Outline for this Part

1. **Keyword table search**
2. Query-by-table

Motivation for Keyword Table Search

Many queries ask for a list of things.

Europe's largest cities Cities ranked 1 to 100

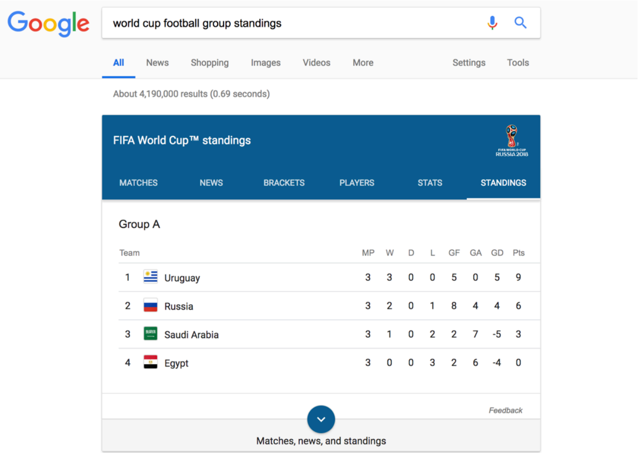
Rank	City	Population
1	MOSKVA (Moscow)	8,297,000
2	LONDON	7,074,000
3	St Petersburg	4,678,000
4	BERLIN	3,387,000

96 more rows

City Mayors: The 500 largest European cities (1 to 100)
www.citymayors.com/features/euro_cities1.html

Motivation for Keyword Table Search

Return a table instead as a result.



	Team	MP	W	D	L	GF	GA	GD	Pts
1	Uruguay	3	3	0	0	5	0	5	9
2	Russia	3	2	0	1	8	4	4	6
3	Saudi Arabia	3	1	0	2	2	7	-5	3
4	Egypt	3	0	0	3	2	6	-4	0

Keyword Table Search

Definition

Given a keyword query, the task of returning a ranked list of tables as results is called *keyword table search*.

Approaches

- Baseline: treating tables as documents
- Challenges:
    - Signals that work well for documents don't necessarily apply here (e.g., term proximity)
    - Variations in table layout or terminology change the semantics significantly

Approaches

- Unsupervised methods
  - Build a document-based representation for each table, then employ conventional document retrieval methods (Cafarella et al., 2008, 2009)
- Supervised methods
  - Describe query-table pairs using a set of features, then employ supervised machine learning (i.e., learning-to-rank) (Bhagavatula et al., 2013)

Unsupervised methods

- Single-field document representation
  - All table content, no structure
- Multi-field document representation
  - Separate document fields for various table elements (embedding document's title, section title, table caption, table body, and table headings)

The Anatomy of a Relational Table

Figure: Illustration of table elements in a web table: table page title (T_p), table caption (T_c), table headings (T_H), table cell ($T_{[i,j]}$), table row ($T_{[i,:]}$), table column ($T_{[:,j]}$), and table entities (T_E).

Supervised Methods

- Three groups of features
- **Query features**
  - #query terms, query IDF scores
- **Table features**
  - Table properties: #rows,, #cols, #empty cells, etc.
  - Embedding documents: link structure, number of tables, etc
- **Query-table features**
  - Query terms found in different table elements, LM score, etc

Features for Table Retrieval

Query features		**Source**
QLEN	Number of query terms	(Tyree et al., 2011)
IDF_f	Sum of query IDF scores in field f	(Qin et al., 2010)
Table features		
#rows	The number of rows in the table	(Cafarella et al., 2008; Bhagavatula et al., 2013)
#cols	The number of columns in the table	(Cafarella et al., 2008; Bhagavatula et al., 2013)
#of NULLs in table	The number of empty table cells	(Cafarella et al., 2008; Bhagavatula et al., 2013)
PMI	The ACSDb-based schema coherency score	(Cafarella et al., 2008)
inLinks	Number of in-links to the page embedding the table	(Bhagavatula et al., 2013)
outLinks	Number of out-links from the page embedding the table	(Bhagavatula et al., 2013)
pageViews	Number of page views	(Bhagavatula et al., 2013)
tableImportance	Inverse of number of tables on the page	(Bhagavatula et al., 2013)
tablePageFraction	Ratio of table size to page size	(Bhagavatula et al., 2013)

Features for Table Retrieval (2)

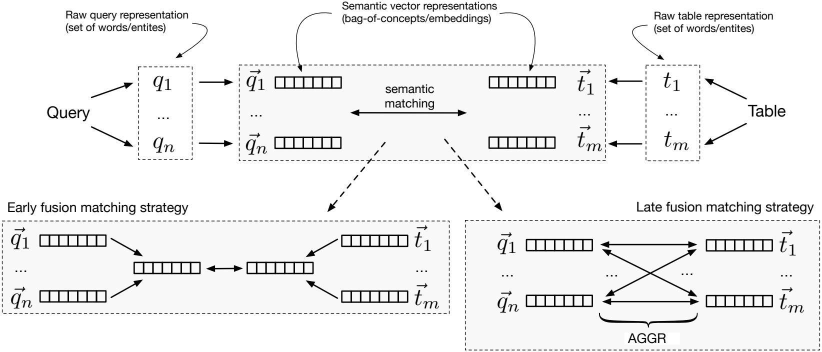
Query-table features		
#hitsLC	Total query term frequency in the leftmost column cells	(Cafarella et al., 2008)
#hitsSLC	Total query term frequency in second-to-leftmost column cells	Cafarella et al. (2008)
#hitsB	Total query term frequency in the table body	(Cafarella et al., 2008)
qInPgTitle	Ratio of the number of query tokens found in page title to total number of tokens	(Bhagavatula et al., 2013)
qInTableTitle	Ratio of the number of query tokens found in table title to total number of tokens	(Bhagavatula et al., 2013)
yRank	Rank of the table's Wikipedia page in Web search engine results for the query	(Bhagavatula et al., 2013)
MLM similarity	Language modeling score between query and multi-field document repr. of the table	(Chen et al., 2016)

Ad hoc table retrieval (Zhang and Balog, 2018)

They perform semantic matching between queries and tables for keyword table search.

1. Content extraction
   - The "raw" content of a query/table is represented as a set of terms, which can be words or entities
2. Semantic representations
   - Each of the raw terms is mapped to a semantic vector representation
   - Bag-of-concepts, word and graph embeddings
3. Similarity measures

Illustration of Semantic Matching



Evaluation

- **Wikipedia Table Corpus:** it contains 1.65M high-quality tables
- **DBpedia:** 4.6M entities
- Test queries sampled from two sources: QS-1, QS-2
- Rank-based evaluation (NDCG@5, 10, 15, 20)

QS-1 (Cafarella et al., 2009)	**QS-2 (Venetis et al., 2011)**
video games	asian coutries currency
us cities	laptops cpu
kings of africa	food calories
economy gdp	guitars manufacturer
fifa world cup winners	clothes brand

Relevance Assessments

- Collected via crowdsourcing
  - Pooling to depth 20, 3120 query-table pairs in total
- Assessors are presented with the following scenario: *Imagine that your task is to create a new table on the query topic.*
- A table is ...
  - **Non-relevant** (0): if it is unclear what it is about or it about a different topic
  - **Relevant** (1): if some cells or values could be used from it
  - **Highly relevant** (2): if large blocks or several values could be used from it
- Resources: https://github.com/iai-group/www2018-table

Results

Method	NDCG@5	NDCG@10	NDCG@15	NDCG@20
Single-field document ranking	0.4315	0.4344	0.4586	0.5254
Multi-field document ranking	0.4770	0.4860	0.5170	0.5473
WebTable (Cafarella et al., 2008)	0.2831	0.2992	0.3311	0.3726
WikiTable (Bhagavatula et al., 2013)	0.4903	0.4766	0.5062	0.5206
LTR baseline (Zhang and Balog, 2018)	0.5527	0.5456	0.5738	0.6031
STR (Zhang and Balog, 2018)	**0.5951**	**0.6293**†	**0.6590**‡	**0.6825**†

Take-away Points for Keyword Table Search

- Standard document-based approaches can still be used, but the requirements are different
- Feature-based methods with semantic similarity provide solid performance
- The problem is not yet solved
  - Existing methods assume a specific type of table
  - It is also implicitly assumed that the answer should be a table (automatic query classification would be needed)

Outline for this Part

1. ~~Keyword table search~~
2. **Query-by-table**

Motivation for Search by Table

The input table can be the query.

MotoGP World Standing 2017

Pos.	Rider	Bike	Points
1	Marc MARQUEZ	Honda	282
2	Andrea DOVIZIOSO	Ducati	261
3	Maverick VINALES	Yamaha	226
4	Valentino ROSSI	Yamaha	197

Related tables

MotoGP 2016 Championship Final Standing

Pos.	Rider	Bike	Nation	Points
1	Marc MARQUEZ	Honda	SPA	298
2	Valentino ROSSI	Yamaha	ITA	249
3	Jorge LORENZO	Yamaha	SPA	233
4	Maverick VINALES	Suzuki	SPA	202
...	...	...	...	...

Grand Prix motorcycle racing World champions

Rank	Rider	Country	Period	Total
1	Giacomo Agostini	Italy	1966-1975	15
2	Angel Nieto	Spain	1969-1984	13
3	Valentino Rossi	Italy	1997-2009	9
3	Mike Hailwood	UK	1961-1967	9
...	...	...	...	...

Query-by-Table

Definition

Given an input table, the task of returning related tables is referred to as *search by table* or *query-by-table*.

Overview of Approaches

- Based on the goal:
  - to be presented to the user to answer her information need (Das Sarma et al., 2012; Limaye et al., 2010)
  - to serve as an intermediate step that feeds into other tasks, like table augmentation (Ahmadov et al., 2015; Lehmberg et al., 2015)
- Based on the method used:
  - Using certain table elements as a keyword query (Lehmberg et al., 2015; Ahmadov et al., 2015)
  - Dividing tables into various elements (such as table caption, table entities, column headings, cell values), then computing element-level similarities (Das Sarma et al., 2012; Yakout et al., 2012; Nguyen et al., 2015)

Table Elements Utilized

Application	T_E	T_H	$T_{[:,j]}$	T_p	$T_{[i,j]}$
Data completion (Ahmadov et al., 2015)	✓	✓			
Relation join (Lehmberg et al., 2015)		✓			
Schema complement (Das Sarma et al., 2012)	✓	✓			
Entity complement (Das Sarma et al., 2012)	✓				
InfoGather (Yakout et al., 2012)		✓	✓	✓	✓
Diverse table search (Nguyen et al., 2015)		✓			✓
Table cell retrieval (Limaye et al., 2010)			✓		✓
Table union search (Nargesian et al., 2018)	✓		✓		✓

Finding Tables for Data Augmentation

- Find related tables for augmenting the input table with
  - additional rows (*entity complement*) and columns (*schema complement*) (Das Sarma et al., 2012)
  - additional attributes (Lehmberg et al., 2015)
- Augmentation based on various elements of the input table: column headings, augmentation by example, and column heading discovery (Yakout et al., 2012)
- Augmentation approaches will be detailed later in Part V

Query-by-Table == Table Matching

Query-by-table boils down to **table matching**, which is commonly performed as either of the following two methods:

- Extracting a keyword query (from various table elements) and scoring tables against that query
- Splitting tables into various elements and performing element-wise matching
  - Ad hoc similarity measures, tailor-made for each table element
  - Lacking a principled way of combining element-level similarities
  - Matching elements of different types have not been explored

Semantic Matching for Query-by-Table (Zhang and Balog, 2019)

Build on idea in (Zhang and Balog, 2018):

- Represent table elements in multiple semantic spaces
- Measure element-level similarity in each of the semantic spaces
- Combine the element-level similarities in a discriminative learning framework

Table Matching Results (Zhang and Balog, 2019)

Method	NDCG@5	NDCG@10
Keyword-based search using T_E	0.2001	0.1998
Keyword-based search using T_H	0.2318	0.2527
Keyword-based search using T_c	0.1369	0.1419
Mannheim Search Join Engine (Lehmberg et al., 2015)	0.3298	0.3131
Schema complement (Das Sarma et al., 2012)	0.3389	0.3418
Entity complement (Das Sarma et al., 2012)	0.2986	0.3093
Nguyen et al. (2015)	0.2875	0.3007
InfoGather (Yakout et al., 2012)	**0.4530**	**0.4686**
HCF-1 (Zhang and Balog, 2019)	0.5382	0.5542
HCF-2 (Zhang and Balog, 2019)	**0.5895**	**0.6050**
CRAB-1 (Zhang and Balog, 2019)	0.5578	0.5672
CRAB-2 (Zhang and Balog, 2019)	**0.6172**	**0.6267**

Table: Evaluation results reported in (Zhang and Balog, 2019). HCF combines features from existing approaches, while CRAB is based on semantic matching between corresponding table elements. HCF-1 and CRAB-1 employs only table similarity features, HCF-2 and CRAB-2 also incorporate table features.

HCF-1 Features (Zhang and Balog, 2019)

Element / Feature	**Reference**
Page title ($\tilde{T}_p \leftrightarrow T_p$)	
InfoGather page title IDF similarity score	(Yakout et al., 2012)
Table headings ($\tilde{T}_H \leftrightarrow T_H$)	
MSJE heading matching score	Lehmberg et al. (2015)
Schema complement schema benefit score	(Das Sarma et al., 2012)
InfoGather heading-to-heading similarity	(Yakout et al., 2012)
Nguyen et al. heading similarity	(Nguyen et al., 2015)
Table data ($\tilde{T}_D \leftrightarrow T_D$)	
InfoGather column-to-column similarity	(Yakout et al., 2012)
InfoGather table-to-table similarity	(Yakout et al., 2012)
Nguyen et al. table data similarity	(Nguyen et al., 2015)
Table entities ($\tilde{T}_E \leftrightarrow T_E$)	
Entity complement entity relatedness score	(Das Sarma et al., 2012)
Schema complement entity overlap score	(Das Sarma et al., 2012)

Take-away Points for Query-by-Table

- Query-by-table boils down to table matching
- Table element specific feature design can be replaced by semantic matching using embeddings
- Relevance criteria is task specific, depends on how tables will be utilized in downstream processing

Bibliography I

Ahmad Ahmadov, Maik Thiele, Julian Eberius, Wolfgang Lehner, and Robert Wrembel. Towards a hybrid imputation approach using web tables. In *Proceedings of the IEEE 2nd International Symposium on Big Data Computing*, BDC '15, pages 21–30. IEEE, 2015. ISBN 978-0-7695-5696-3.

Chandra Sekhar Bhagavatula, Thanapon Noraset, and Doug Downey. Methods for exploring and mining tables on wikipedia. In *Proceedings of the ACM SIGKDD Workshop on Interactive Data Exploration and Analytics*, IDEA '13, pages 18–26, New York, NY, USA, 2013. ACM.

Michael J. Cafarella, Alon Halevy, Daisy Zhe Wang, Eugene Wu, and Yang Zhang. Webtables: Exploring the power of tables on the web. *Proc. VLDB Endow.*, 1(1):538–549, August 2008. ISSN 2150-8097.

Michael J. Cafarella, Alon Halevy, and Nodira Khoussainova. Data integration for the relational web. *Proc. VLDB Endow.*, 2(1):1090–1101, August 2009. ISSN 2150-8097.

Jing Chen, Chenyan Xiong, and Jamie Callan. An empirical study of learning to rank for entity search. In *Proc. of SIGIR '16*, pages 737–740, 2016.

Anish Das Sarma, Lujun Fang, Nitin Gupta, Alon Halevy, Hongrae Lee, Fei Wu, Reynold Xin, and Cong Yu. Finding related tables. In *Proceedings of the 2012 ACM SIGMOD International Conference on Management of Data*, SIGMOD '12, pages 817–828, New York, NY, USA, 2012. ACM.

Bibliography II

Oliver Lehmberg, Dominique Ritze, Petar Ristoski, Robert Meusel, Heiko Paulheim, and Christian Bizer. The mannheim search join engine. *Web Semant.*, 35(P3):159–166, December 2015. ISSN 1570-8268.

Girija Limaye, Sunita Sarawagi, and Soumen Chakrabarti. Annotating and searching web tables using entities, types and relationships. *Proc. VLDB Endow.*, 3(1-2):1338–1347, September 2010.

Fatemeh Nargesian, Erkang Zhu, Ken Q. Pu, and Renée J. Miller. Table union search on open data. *Proc. VLDB Endow.*, 11(7):813–825, March 2018. ISSN 2150-8097.

Thanh Tam Nguyen, Quoc Viet Hung Nguyen, Weidlich Matthias, and Aberer Karl. Result selection and summarization for web table search. In *Proceedings of the 31st International Conference on Data Engineering*, ISDE '15, pages 231–242, 2015.

Tao Qin, Tie-Yan Liu, Jun Xu, and Hang Li. Letor: A benchmark collection for research on learning to rank for information retrieval. *Inf. Retr.*, 13(4):346–374, August 2010. ISSN 1386-4564.

Stephen Tyree, Kilian Q. Weinberger, Kunal Agrawal, and Jennifer Paykin. Parallel boosted regression trees for web search ranking. In *Proceedings of the 20th International Conference on World Wide Web*, WWW '11, pages 387–396, New York, NY, USA, 2011. ACM.

Petros Venetis, Alon Halevy, Jayant Madhavan, Marius Pașca, Warren Shen, Fei Wu, Gengxin Miao, and Chung Wu. Recovering semantics of tables on the web. *Proc. VLDB Endow.*, 4 (9):528–538, June 2011. ISSN 2150-8097.

Bibliography III

Mohamed Yakout, Kris Ganjam, Kaushik Chakrabarti, and Surajit Chaudhuri. Infogather: Entity augmentation and attribute discovery by holistic matching with web tables. In *Proceedings of the 2012 ACM SIGMOD International Conference on Management of Data*, SIGMOD '12, pages 97–108, New York, NY, USA, 2012. ACM.

Shuo Zhang and Krisztian Balog. Ad hoc table retrieval using semantic similarity. In *Proceedings of The Web Conference*, WWW '18, pages 1553–1562, 2018.

Shuo Zhang and Krisztian Balog. Recommending related tables. *CoRR*, abs/1907.03595, 2019.