TREC CAR 2017 Run Summary

CUISER

<table>
<thead>
<tr>
<th>Team</th>
<th>Task</th>
<th>Run type</th>
<th>Ranking methods</th>
</tr>
</thead>
<tbody>
<tr>
<td>CUIS</td>
<td>entity</td>
<td>automatic</td>
<td>BM25 Sequential Dependence Model</td>
</tr>
</tbody>
</table>

Features

- Trained on train-v1.5

Description

Replace para_id in passage ranking results by article_id

CUISPR

<table>
<thead>
<tr>
<th>Team</th>
<th>Task</th>
<th>Run type</th>
<th>Ranking methods</th>
</tr>
</thead>
<tbody>
<tr>
<td>CUIS</td>
<td>passage</td>
<td>automatic</td>
<td>BM25 Sequential Dependence Model</td>
</tr>
</tbody>
</table>

Features

- Trained on train-v1.5

Description

Scoring pass

1. BM25 by Lucene 6 scoring pass
2. a slightly modified sequential dependence model

ECNU-runONE
<table>
<thead>
<tr>
<th>Team</th>
<th>Task</th>
<th>Run type</th>
<th>Ranking methods</th>
</tr>
</thead>
<tbody>
<tr>
<td>ECNU</td>
<td>passage</td>
<td>automatic</td>
<td>BM25</td>
</tr>
</tbody>
</table>

**Features**

- Trained on `train-v1.5`
- Uses learning-to-rank.

**Description**

First, we use lucene to select candidate paragraphs, and then we use bm25 score and word matching as features to training a ranking model with Ranklib.

**treccarict**

<table>
<thead>
<tr>
<th>Team</th>
<th>Task</th>
<th>Run type</th>
<th>Ranking methods</th>
</tr>
</thead>
<tbody>
<tr>
<td>ICTNET</td>
<td>passage</td>
<td>automatic</td>
<td>BM25</td>
</tr>
</tbody>
</table>

**Features**

- Uses a dump of Wikipedia
- Unsupervised

**Description**

Use BM25 for retrieval.

**mpi-nn4_pos_hperc**

<table>
<thead>
<tr>
<th>Team</th>
<th>Task</th>
<th>Run type</th>
<th>Ranking methods</th>
</tr>
</thead>
<tbody>
<tr>
<td>MPIID5</td>
<td>passage</td>
<td>automatic</td>
<td>BM25</td>
</tr>
</tbody>
</table>
Features

- Uses data provided as `unprocessedtrain` or `halfwiki`
- Uses pre-trained word embeddings
- Is trained with data provided as `train-v1.5`

Description

PACRR variant on BM25 candidate list.

**mpi6-nn6_pos**

<table>
<thead>
<tr>
<th>Team</th>
<th>Task</th>
<th>Run type</th>
<th>Ranking methods</th>
</tr>
</thead>
<tbody>
<tr>
<td>MPIID5</td>
<td>passage</td>
<td>automatic</td>
<td>BM25</td>
</tr>
</tbody>
</table>

Features

- Uses data provided as `unprocessedtrain` or `halfwiki`
- Uses pre-trained word embeddings
- Is trained with data provided as `train-v1.5`

Description

PACRR variant on BM25 candidate list.

**mpi6-nn6_pos_tprob**

<table>
<thead>
<tr>
<th>Team</th>
<th>Task</th>
<th>Run type</th>
<th>Ranking methods</th>
</tr>
</thead>
<tbody>
<tr>
<td>MPIID5</td>
<td>passage</td>
<td>automatic</td>
<td>BM25</td>
</tr>
</tbody>
</table>

Features

- Uses data provided as `unprocessedtrain` or `halfwiki`
- Uses pre-trained word embeddings
• Is trained with data provided as train-v1.5

Description

PACRR variant on BM25 candidate list.

nyudl-ds

<table>
<thead>
<tr>
<th>Team</th>
<th>Task</th>
<th>Run type</th>
<th>Ranking methods</th>
</tr>
</thead>
<tbody>
<tr>
<td>NYUDL</td>
<td>passage</td>
<td>automatic</td>
<td>BM25</td>
</tr>
</tbody>
</table>

Features

• Uses data provided as unprocessed\textit{train} or \textit{halfwiki}
• Uses pre-trained word embeddings
• Uses neural network technology
• Uses learning-to-rank
• Is trained with data provided as \textit{train-v1.5}

Description

Simple Document Classifier using avg word embeddings in the documents as document vector and last hidden state of an LSTM as query vector. 2-layer feed forward neural net to select which documents are relevant given a query.

nyudl-qr

<table>
<thead>
<tr>
<th>Team</th>
<th>Task</th>
<th>Run type</th>
<th>Ranking methods</th>
</tr>
</thead>
<tbody>
<tr>
<td>NYUDL</td>
<td>passage</td>
<td>automatic</td>
<td>BM25</td>
</tr>
</tbody>
</table>

Features

• Uses a dump of Wikipedia
• Uses pre-trained word embeddings
- Uses neural network technology
- Uses learning-to-rank
- Uses other supervised machine learning method
- Is trained with data provided as train-v1.5

**Description**

Query reformulation with deep reinforcement learning.

**nyudl-qrds**

<table>
<thead>
<tr>
<th>Team</th>
<th>Task</th>
<th>Run type</th>
<th>Ranking methods</th>
</tr>
</thead>
<tbody>
<tr>
<td>NYUDL</td>
<td>passage</td>
<td>automatic</td>
<td>BM25 Neural network classifier</td>
</tr>
</tbody>
</table>

**Features**

- Uses a dump of Wikipedia
- Uses pre-trained word embeddings
- Uses neural network technology
- Uses learning-to-rank
- Is trained with data provided as train-v1.5

**Description**

Query reformulation using reinforcement learning + Lucene + Neural Net Classifier to select documents

**UNH-benchmarkY1test.bm25**

<table>
<thead>
<tr>
<th>Team</th>
<th>Task</th>
<th>Run type</th>
<th>Ranking methods</th>
</tr>
</thead>
<tbody>
<tr>
<td>TREMA-UNH</td>
<td>passage</td>
<td>automatic</td>
<td>BM25</td>
</tr>
</tbody>
</table>
Features

- Unsupervised

Description

BM25 run using the concatenation of heading, parent headings and page title as keyword query

UNH-benchmarkY1test.expan

<table>
<thead>
<tr>
<th>Team</th>
<th>Task</th>
<th>Run type</th>
<th>Ranking methods</th>
</tr>
</thead>
<tbody>
<tr>
<td>TREMA-UNH</td>
<td>passage</td>
<td>automatic</td>
<td>BM25</td>
</tr>
<tr>
<td></td>
<td></td>
<td></td>
<td>Query expansion (entities and section headings)</td>
</tr>
</tbody>
</table>

Features

- Uses a linguistic graph such as Wordnet
- Uses entity linking
- Uses data provided as ‘unprocessedtrain’ or ‘halfwiki’
- Uses other supervised machine learning method
- Is trained with data provided as train-v1.5
- Is trained with data provided as test200

Description

BM25 run using the concatenation of heading, parent headings and page title as keyword query. In addition query expansion with two sources: 1. the query is entity linked with TagMe. Terms from the first paragraph of the entity are used for expansion (like RM3) 2. if the same heading is contained in another article, then expansion terms from these sections will be used for expansion (like RM3).

This method is in spirit of the WikiKreator system. Balancing parameters are manually adjusted on test200
**top100-c-pr-bm25**

<table>
<thead>
<tr>
<th>Team</th>
<th>Task</th>
<th>Run type</th>
<th>Ranking methods</th>
</tr>
</thead>
<tbody>
<tr>
<td>TREMA-UNH</td>
<td>entity</td>
<td>automatic</td>
<td>BM25</td>
</tr>
<tr>
<td></td>
<td></td>
<td></td>
<td>PageRank</td>
</tr>
</tbody>
</table>

**Features**

- Uses a knowledge graph such as DBpedia, Freebase, etc.
- Uses data provided as 'unprocessedtrain' or 'halfwiki'
- Unsupervised
- Is trained with data provided as benchmarkY1train

**Description**

Ranks entities by degree centrality on a sub-graph that is extracted as follows: 1. edges in KG are associated with paragraph-long text 2. A BM25 model is used to retrieve edges in response to the query 3. edges are weighted according to their frequency 4. PageRank on this weighted graph is used to rank entities Support paragraphs are taken from the paragraph associated with the entity’s highest ranking edge.

**top100-rr-marg-bm25**

<table>
<thead>
<tr>
<th>Team</th>
<th>Task</th>
<th>Run type</th>
<th>Ranking methods</th>
</tr>
</thead>
<tbody>
<tr>
<td>TREMA-UNH</td>
<td>entity</td>
<td>automatic</td>
<td>BM25</td>
</tr>
<tr>
<td></td>
<td></td>
<td></td>
<td>Degree centrality</td>
</tr>
</tbody>
</table>

**Features**

- Uses a knowledge graph such as DBpedia, Freebase, etc.
- Uses entity linking
- Uses data provided as unprocessedtrain or halfwiki
- Does not make use of any training data (unsupervised)
- Is trained with data provided as train-v1.5
• Is trained with data provided as benchmarkY1train

Description

Ranks entities by degree centrality on a sub-graph that is extracted as follows:
1. edges in KG are associated with paragraph-long text.
2. A BM25 model is used to retrieve edges in response to the query.
3. edges are weighted according to their reciprocal rank.
4. DegreeCentrality on this weighted graph is used to rank entities.
Support paragraphs are taken from the paragraph associated with the entity’s highest ranking edge.

top100-sc-ppr-bm25

<table>
<thead>
<tr>
<th>Team</th>
<th>Task</th>
<th>Run type</th>
<th>Ranking methods</th>
</tr>
</thead>
</table>
| TREMA-UNH    | entity| automatic| BM25
Personalized|
|              |       |          | PageRank                |

Features

• Uses a knowledge graph such as DBpedia, Freebase, etc.
• Uses entity linking
• Uses data provided as 'unprocessedtrain' or 'halfwiki'
• Does not make use of any training data (unsupervised)
• Is trained with data provided as train-v1.5
• Is trained with data provided as benchmarkY1train

Description

Ranks entities by degree centrality on a sub-graph that is extracted as follows:
1. edges in KG are associated with paragraph-long text.
2. A BM25 model is used to retrieve edges in response to the query.
3. edges are weighted according to their frequency.
4. seed nodes are retrieved from an entity index of unprocessedtrain using BM25.
5. PersonalizedPageRank on this weighted graph with seed nodes is used to rank entities.
Support paragraphs are taken from the paragraph associated with the entity’s highest ranking edge.
## UTDHLTRIAR

<table>
<thead>
<tr>
<th>Team</th>
<th>Task</th>
<th>Run type</th>
<th>Ranking methods</th>
</tr>
</thead>
<tbody>
<tr>
<td>UTDHLTRI</td>
<td>passage</td>
<td>automatic</td>
<td>BM25</td>
</tr>
<tr>
<td></td>
<td></td>
<td></td>
<td>TF-IDF</td>
</tr>
<tr>
<td></td>
<td></td>
<td></td>
<td>Dirichlet Language Model</td>
</tr>
</tbody>
</table>

### Features
- Uses a knowledge graph such as DBpedia, Freebase, etc.
- Uses entity linking
- Uses learning-to-rank
- Is trained with data provided as test200
- Is trained with data provided as benchmarkY1train

### Description
1. Initial Search
   1. Parse the article title and section names to use in part of query for Lucene search using BM25
   2. Use entities extracted from Dbpedia long abstracts data set from Dbpedia spotlight tagging to augment query
2. Re-Ranking
   1. Use Ranklib’s Ada Rank implementation to re-rank paragraphs

## UTDHLTRINN20

<table>
<thead>
<tr>
<th>Team</th>
<th>Task</th>
<th>Run type</th>
<th>Ranking methods</th>
</tr>
</thead>
<tbody>
<tr>
<td>UTDHLTRI</td>
<td>passage</td>
<td>automatic</td>
<td>BM25</td>
</tr>
<tr>
<td></td>
<td></td>
<td></td>
<td>TF-IDF</td>
</tr>
<tr>
<td></td>
<td></td>
<td></td>
<td>Dirichlet Language Model</td>
</tr>
</tbody>
</table>

### Features
- Uses a knowledge graph such as DBpedia, Freebase, etc.
• Uses entity linking
• Uses pre-trained word embeddings
• Uses neural network technology
• Uses learning-to-rank
• Is trained with data provided as train-v1.5

Description
1. Initial Search
   1. Parse the article title and section names to use in part of query for Lucene search using BM25
   2. Use entities extracted from Dbpedia long abstracts data set from Dbpedia spotlight tagging to augment query
2. Re-Ranking
   1. use neural learning to rank model to re-rank paragraphs from Initial Search

UTDHLTRINN50

<table>
<thead>
<tr>
<th>Team</th>
<th>Task</th>
<th>Run type</th>
<th>Ranking methods</th>
</tr>
</thead>
<tbody>
<tr>
<td>UTDHLTRI</td>
<td>passage</td>
<td>automatic</td>
<td>BM25</td>
</tr>
<tr>
<td></td>
<td></td>
<td></td>
<td>TF-IDF</td>
</tr>
<tr>
<td></td>
<td></td>
<td></td>
<td>Dirichlet Language Model</td>
</tr>
</tbody>
</table>

Features

• Uses a knowledge graph such as DBpedia, Freebase, etc.
• Uses entity linking
• Uses pre-trained word embeddings
• Uses neural network technology
• Uses learning-to-rank
• Is trained with data provided as train-v1.5

Description

1. Initial Search
1. Parse the article title and section names to use in part of query for Lucene search using BM25
2. Use entities extracted from Dbpedia long abstracts data set from Dbpedia spotlight tagging to augment query

2. Re-Ranking
   1. use neural learning to rank model to re-rank paragraphs from Initial Search