Fielded Sequential Dependence Model for Ad-Hoc Entity Retrieval in the Web of Data

Nikita Zhiltsov\textsuperscript{1,2} \quad Alexander Kotov\textsuperscript{3} \quad Fedor Nikolaev\textsuperscript{3}

\textsuperscript{1}Kazan Federal University
\textsuperscript{2}Textocat
\textsuperscript{3}Textual Data Analytics Lab, Department of Computer Science, Wayne State University
Overview

Entities

Entity Representation

Fielded Sequential Dependence Model

Parameter Estimation

Results

Conclusion
Knowledge Graphs

- Freebase
- Yago
- DBpedia
- Facebook’s Entity Graph
- Microsoft’s Satori
- Google’s Knowledge Graph
- OpenIE (Reverb, OLLIE)
Linked Open Data (LOD) Cloud
**Entities**

- Material objects or concepts in the real world or fiction (e.g. people, movies, conferences etc.)
- Are connected with other entities by relations (e.g. hasGenre, actedIn, isPCmemberOf etc.)
- Subject-Predicate-Object (SPO) triple: subject=entity; object=entity (or primitive data value); predicate=relationship between subject and object
- Many SPO triples → knowledge graph
DBPedia entity page example

About: **Barack Obama**

An Entity of Type: **office_holder**, from Named Graph: [http://dbpedia.org](http://dbpedia.org), within Data Space: [dbpedia.org](http://dbpedia.org)

Barack Hussein Obama II (ˈbærək hʊˈseɪn əˌbɑːmaː; born August 4, 1961) is the 44th and current President of the United States, and the first African American to hold the office. Born in Honolulu, Hawaii, Obama is a graduate of Columbia University and Harvard Law School, where he served as president of the Harvard Law Review. He was a community organizer in Chicago before earning his law degree.

**rdfs:comment**
- Barack Hussein Obama II (ˈbærək hʊˈseɪn əˌbɑːmaː; born August 4, 1961) is the 44th and current President of the United States, and the first African American to hold the office. Born in Honolulu, Hawaii, Obama is a graduate of Columbia University and Harvard Law School, where he served as president of the Harvard Law Review. He was a community organizer in Chicago before earning his law degree.

**rdfs:label**
- Barack Obama
- BarackObama

**is dbpedia-owl:author of**
- dbpedia:Dreams_from_My_Father
- dbpedia:The_Audacity_of_Hope
- dbpedia:Of_Thee_I_Sing_(book)

**is dbpedia-owl:child of**
- dbpedia:Ann_Dunham
- dbpedia:Lolo_Soetoro
- dbpedia:Barack_Obama_Sr.

**dc:description**
- American politician, 44th President of the United States
- American politician, 44th President of the United States

**dcterms:subject**
- category:African-American_academics
- category:American_civil_rights_lawyers
- category:Illinois_State_Senators
- category:United_Church_of_Christ_members
- category:1961_births
- category:20th-century_American_writers
- category:21st-century_American_writers
- category:African-American_Christians
Entity Retrieval from Knowledge Graph(s)

- Graph KBs are perfectly suited for addressing the information needs that aim at finding specific objects (entities) rather than documents.

- Given the user’s information need expressed as a keyword query, retrieve a relevant set of objects from the knowledge graph(s).
Typical ERWD tasks

- **Entity Search**
  Queries refer to a particular entity.
  - “Ben Franklin”
  - “England football player highest paid”
  - “Einstein Relativity theory”

- **List Search**
  Complex queries with several relevant entities.
  - “US presidents since 1960”
  - “animals lay eggs mammals”

- **Question Answering**
  Queries are questions in natural language.
  - “Who is the mayor of Santiago?”
  - “For which label did Elvis record his first album?”
FUNDAMENTAL PROBLEMS IN ERWD

▶ Designing effective and concise entity representations
  - Pound, Mika et al. Ad-hoc Object Retrieval in the Web of Data, WWW’10
  - Blanco, Mika et al. Effective and Efficient Entity Search in RDF Data, ISWC’11
  - Neumayer, Balog et al. On the Modeling of Entities for Ad-hoc Entity Search in the Web of Data, ECIR’12

▶ Developing accurate retrieval models
  - Mostly adaptations of standard unigram bag-of-words retrieval models, such as BM25F, MLM
Overview

Entities

Entity Representation

Fielded Sequential Dependence Model

Parameter Estimation

Results

Conclusion
**ENTITY DOCUMENT**

An entity is represented as a structured (multi-fielded) document:

- **names**: Conventional names of the entities, such as the name of a person or the name of an organization
- **attributes**: All entity properties, other than names
- **categories**: Classes or groups, to which the entity has been assigned
- **similar entity names**: Names of the entities that are very similar or identical to a given entity
- **related entity names**: Names of the entities that are part of the same RDF triple
## Entity document example

Multi-fielded entity document for the entity *Barack Obama*.

<table>
<thead>
<tr>
<th>Field</th>
<th>Content</th>
</tr>
</thead>
<tbody>
<tr>
<td>names</td>
<td>barack obama barack hussein obama ii</td>
</tr>
<tr>
<td>attributes</td>
<td>44th current president united states</td>
</tr>
<tr>
<td></td>
<td>birth place honolulu hawaii</td>
</tr>
<tr>
<td>categories</td>
<td>democratic party united states senator</td>
</tr>
<tr>
<td></td>
<td>nobel peace prize laureate christian</td>
</tr>
<tr>
<td>similar entity names</td>
<td>barack obama jr barak hussein obama</td>
</tr>
<tr>
<td></td>
<td>barack h obama ii</td>
</tr>
<tr>
<td>related entity names</td>
<td>spouse michelle obama illinois state</td>
</tr>
<tr>
<td></td>
<td>predecessor george walker bush</td>
</tr>
</tbody>
</table>
Overview

Entities

Entity Representation

Fielded Sequential Dependence Model

Parameter Estimation

Results

Conclusion
Motivation

Previous research in ad-hoc IR has focused on two major directions:

- unigram bag-of-words retrieval models for multi-fielded documents
  - Ogilvie and Callan. Combining Document Representations for Known-item Search, SIGIR’03
  - Robertson et al. Simple BM25 Extension to Multiple Weighted Fields, CIKM’04

- retrieval models incorporating term dependencies
  - Metzler and Croft. A Markov Random Field Model for Term Dependencies, SIGIR’05
  - Huston and Croft. A Comparison of Retrieval Models using Term Dependencies, CIKM’14

Goal: to develop a retrieval model that captures both document structure and term dependencies
MLM

\[ P(Q|D) = \prod_{q_i \in Q} P(q_i|\theta_D)^{t_f(q_i)}, \]

where

\[ P(q_i|\theta_D) = \sum_j w_j P(q_i|\theta_D^j). \]
SDM

Ranks w.r.t. $P_{\Lambda}(D|Q) = \sum_{i \in \{T, U, O\}} \lambda_i f_i(Q, D)$

Potential function for unigrams is QL:

$$f_T(q_i, D) = \log P(q_i|\theta_D) = \log \frac{tf_{q_i,D} + \mu \frac{cf_{q_i}}{|C|}}{|D| + \mu}$$
FSDM ranking function

FSDM incorporates document structure and term dependencies with the following ranking function:

\[
P^\text{rank}_\Lambda(D|Q) = \lambda_T \sum_{q \in Q} \tilde{f}_T(q_i, D) + \lambda_O \sum_{q \in Q} \tilde{f}_O(q_i, q_{i+1}, D) + \lambda_U \sum_{q \in Q} \tilde{f}_U(q_i, q_{i+1}, D)
\]

Separate MLMs for bigrams and unigrams give FSDM the flexibility to adjust the document scoring depending on the query type.

MLM is a special case of FSDM, when \( \lambda_T = 1, \lambda_O = 0, \lambda_U = 0 \).
FSDM ranking function

FSDM incorporates document structure and term dependencies with the following ranking function:

\[
P_A(D|Q) = \lambda_T \sum_{q \in Q} \tilde{f}_T(q_i, D) + \lambda_O \sum_{q \in Q} \tilde{f}_O(q_i, q_{i+1}, D) + \lambda_U \sum_{q \in Q} \tilde{f}_U(q_i, q_{i+1}, D)
\]

Separate MLMs for bigrams and unigrams give FSDM the flexibility to adjust the document scoring depending on the query type.

MLM is a special case of FSDM, when \( \lambda_T = 1, \lambda_O = 0, \lambda_U = 0 \)
FSDM ranking function

FSDM incorporates document structure and term dependencies with the following ranking function:

\[
P_\Lambda(D|Q) = \lambda_T \sum_{q \in Q} \tilde{f}_T(q_i, D) + \lambda_O \sum_{q \in Q} \tilde{f}_O(q_i, q_{i+1}, D) + \lambda_U \sum_{q \in Q} \tilde{f}_U(q_i, q_{i+1}, D)
\]

Separate MLMs for bigrams and unigrams give FSDM the flexibility to adjust the document scoring depending on the query type

MLM is a special case of FSDM, when \( \lambda_T = 1, \lambda_O = 0, \lambda_U = 0 \)
**FSDM ranking function**

FSDM incorporates document structure and term dependencies with the following ranking function:

$$P_{\Lambda}(D|Q) = \lambda_T \sum_{q \in Q} \tilde{f}_T(q_i, D) + \lambda_O \sum_{q \in Q} \tilde{f}_O(q_i, q_{i+1}, D) + \lambda_U \sum_{q \in Q} \tilde{f}_U(q_i, q_{i+1}, D)$$

Separate MLMs for bigrams and unigrams give FSDM the flexibility to adjust the document scoring depending on the query type.

MLM is a special case of FSDM, when $\lambda_T = 1, \lambda_O = 0, \lambda_U = 0$
**FSDM ranking function**

Potential function for unigrams in case of FSDM:

$$\tilde{f}_T(q_i, D) = \log \sum_j w_j^T P(q_i | \theta_D^j) = \log \sum_j w_j^T \frac{tf_{q_i, D} + \mu_j}{|C_j|} + \mu_j$$

Example

apollo astronauts who walked on the moon
FSDM ranking function

Potential function for unigrams in case of FSDM:

$$\tilde{f}_T(q_i, D) = \log \sum_j w_j^T P(q_i|\theta^j_D) = \log \sum_j w_j^T \frac{tf_{q_i,D} + \mu_j c_{f_q^j}}{|D_j| + \mu_j}$$

Example

apollo astronauts who walked on the moon
category
FSDM ranking function

Potential function for unigrams in case of FSDM:

\[
\tilde{f}_T(q_i, D) = \log \sum_j w_j^T P(q_i | \theta_D^j) = \log \sum_j w_j^T \frac{tf_{q_i,Dj} + \mu_j c_{f_q,|C_j|}}{|Dj| + \mu_j}
\]

Example

apollo astronauts who \textit{walked on the moon} category
\textit{attribute}
PARAMETERS OF FSDM

Overall, FSDM has $3 \times F + 3$ free parameters: $\langle w^T, w^O, w^U, \lambda \rangle$.

Properties of ranking function

1. Linearity with respect to $\lambda$.

   We can apply any linear learning-to-rank algorithm to optimize the ranking function with respect to $\lambda$.

2. Linearity with respect to $w$ of the arguments of monotonic $\tilde{f}(\cdot)$ functions.

   Optimization of the arguments as linear functions with respect to $w$, leads to optimization of each function $\tilde{f}(\cdot)$.
Overview

Entities

Entity Representation

Fielded Sequential Dependence Model

Parameter Estimation

Results

Conclusion
Optimization algorithm

1: $Q \leftarrow$ Training queries
2: for $s \in \{T, O, U\}$ do // Optimize field weights of LMs independently
3: \hspace{1em} $\lambda = e_s$
4: \hspace{1em} $\hat{w}^s \leftarrow CoordAsc(Q, \lambda)$
5: end for
6: $\hat{\lambda} \leftarrow CoordAsc(Q, \hat{w}_T, \hat{w}_O, \hat{w}_U)$ // Optimize $\lambda$

The unit vectors $e_T = (1, 0, 0)$, $e_O = (0, 1, 0)$, $e_U = (0, 0, 1)$ are the corresponding settings of the parameters $\lambda$ in the formula of FSDM ranking function.

⇒ direct optimization w.r.t. target metric, e.g. MAP
Overview

Entities

Entity Representation

Fielded Sequential Dependence Model

Parameter Estimation

Results

Conclusion
Collection

- DBPedia 3.7 was used as a collection in all experiments
- Structured version of on-line encyclopedia Wikipedia
- Provides the descriptions of over 3.5 million entities belonging to 320 classes
Query Sets


<table>
<thead>
<tr>
<th>Query set</th>
<th>Amount</th>
<th>Query types [Pound et al., 2010]</th>
</tr>
</thead>
<tbody>
<tr>
<td>SemSearch ES</td>
<td>130</td>
<td>Entity</td>
</tr>
<tr>
<td>ListSearch</td>
<td>115</td>
<td>Type</td>
</tr>
<tr>
<td>INEX-LD</td>
<td>100</td>
<td>Entity, Type, Attribute, Relation</td>
</tr>
<tr>
<td>QALD-2</td>
<td>140</td>
<td>Entity, Type, Attribute, Relation</td>
</tr>
</tbody>
</table>
**Tuning field weights**

- **Attributes** field is consistently considered to be a very valuable for both unigrams and bigrams.
- The **names** field as well as the **similar entity names** field are highly important for queries aiming at finding named entities.
- Distinguishing **categories** from **related entity names** is particularly important for type queries.
Tuning $\lambda$

(a) SDM

(b) FSDM

- Bigram matches are important for named entity queries.
- Transformation of SDM into FSDM increases the importance of bigram matches, which ultimately improves the retrieval performance, as we will demonstrate next.
## Experimental results

<table>
<thead>
<tr>
<th>Query set</th>
<th>Method</th>
<th>MAP</th>
<th>P@10</th>
<th>P@20</th>
<th>b-pref</th>
</tr>
</thead>
<tbody>
<tr>
<td><strong>SemSearch ES</strong></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>MLM-CA</td>
<td>0.320</td>
<td>0.250</td>
<td>0.179</td>
<td>0.674</td>
<td></td>
</tr>
<tr>
<td>SDM-CA</td>
<td>0.254*</td>
<td>0.202*</td>
<td>0.149*</td>
<td>0.671</td>
<td></td>
</tr>
<tr>
<td>FSDM</td>
<td>0.386†</td>
<td>0.286*</td>
<td>0.204*</td>
<td>0.750*</td>
<td></td>
</tr>
<tr>
<td><strong>ListSearch</strong></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>MLM-CA</td>
<td>0.190</td>
<td>0.252</td>
<td>0.192</td>
<td>0.428</td>
<td></td>
</tr>
<tr>
<td>SDM-CA</td>
<td>0.197</td>
<td>0.252</td>
<td>0.202</td>
<td>0.471*</td>
<td></td>
</tr>
<tr>
<td>FSDM</td>
<td>0.203</td>
<td>0.256</td>
<td>0.203</td>
<td>0.466*</td>
<td></td>
</tr>
<tr>
<td><strong>INEX-LD</strong></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>MLM-CA</td>
<td>0.102</td>
<td>0.238</td>
<td>0.190</td>
<td>0.318</td>
<td></td>
</tr>
<tr>
<td>SDM-CA</td>
<td>0.117*</td>
<td>0.258</td>
<td>0.199</td>
<td>0.335</td>
<td></td>
</tr>
<tr>
<td>FSDM</td>
<td>0.111†</td>
<td>0.263*</td>
<td>0.215*</td>
<td>0.341*</td>
<td></td>
</tr>
<tr>
<td><strong>QALD-2</strong></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>MLM-CA</td>
<td>0.152</td>
<td>0.103</td>
<td>0.084</td>
<td>0.373</td>
<td></td>
</tr>
<tr>
<td>SDM-CA</td>
<td>0.184</td>
<td>0.106</td>
<td>0.090</td>
<td>0.465*</td>
<td></td>
</tr>
<tr>
<td>FSDM</td>
<td>0.195*</td>
<td>0.136*</td>
<td>0.111*</td>
<td>0.466*</td>
<td></td>
</tr>
<tr>
<td><strong>All queries</strong></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>MLM-CA</td>
<td>0.196</td>
<td>0.206</td>
<td>0.157</td>
<td>0.455</td>
<td></td>
</tr>
<tr>
<td>SDM-CA</td>
<td>0.192</td>
<td>0.198</td>
<td>0.155</td>
<td>0.495*</td>
<td></td>
</tr>
<tr>
<td>FSDM</td>
<td>0.231†</td>
<td>0.231*</td>
<td>0.179*</td>
<td>0.517*</td>
<td></td>
</tr>
</tbody>
</table>
Topic-level differences between SDM and FSDM

Topic-level differences in average precision between FSDM and SDM. Positive values indicate FSDM is better.
Overview

Entities

Entity Representation

Fielded Sequential Dependence Model

Parameter Estimation

Results

Conclusion
We proposed Fielded Sequential Dependence Model, a novel retrieval model, which incorporates term dependencies into structured document retrieval.

We proposed a two-stage algorithm to directly optimize the parameters of FSDM with respect to the target retrieval metric.

We experimentally demonstrated that having different field weighting schemes for unigrams and bigrams is effective for different types of ERWD queries.

Experimental evaluation of FSDM on a standard publicly available benchmark showed that it consistently and, in most cases, statistically significantly outperforms MLM and SDM for the task of ERWD.
Code and runs are available at

github.com/teanalab/FieldedSDM

Questions?
Robustness

- FSDM is more robust compared to SDM
- FSDM improves the performance of 50% of the queries with respect to MLM-CA, compared to 45% of the queries improved by SDM
- FSDM decreases the performance of only 26% of the queries, while SDM degrades the performance of 40% of the queries
### Various Levels of Difficulty

<table>
<thead>
<tr>
<th>Level</th>
<th>Model</th>
<th>MAP</th>
<th>P@10</th>
<th>P@20</th>
<th>b-pref</th>
</tr>
</thead>
<tbody>
<tr>
<td><strong>Difficult queries</strong></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>SDM</td>
<td>0.213</td>
<td>0.067</td>
<td>0.042</td>
<td>0.599</td>
<td></td>
</tr>
<tr>
<td>FSDM</td>
<td>0.239</td>
<td>0.065</td>
<td>0.043</td>
<td>0.621</td>
<td></td>
</tr>
<tr>
<td><strong>Medium queries</strong></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>SDM</td>
<td>0.209</td>
<td>0.224</td>
<td>0.165</td>
<td>0.532</td>
<td></td>
</tr>
<tr>
<td>FSDM†</td>
<td>0.264</td>
<td>0.272†</td>
<td>0.191†</td>
<td>0.559†</td>
<td></td>
</tr>
<tr>
<td><strong>Easy queries</strong></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>SDM</td>
<td>0.139</td>
<td>0.298</td>
<td>0.262</td>
<td>0.316</td>
<td></td>
</tr>
<tr>
<td>FSDM†</td>
<td>0.166†</td>
<td>0.345†</td>
<td>0.309†</td>
<td>0.330</td>
<td></td>
</tr>
</tbody>
</table>

Creating sophisticated entity descriptions is not sufficient for answering *difficult queries* in entity retrieval scenario and better capturing the semantics of query terms is required to further improve the precision of FSDM for difficult queries.
Failure Analysis

- SDM errors
  - Overestimation of importance of matches in the fields other than names
    - “city of charlotte”
    - “give me all soccer clubs in the premier league”
    - “us presidents since 1960”

- FSDM errors
  - Neglecting the important query terms
    - “members of the beaux arts trio”
    - “who created goofy”
    - “where is the residence of the prime minister of spain?”
  - Lack of semantic knowledge.
    - “did nicole kidman have any siblings”