Ranking With Neural Network Derived Document Vectors

Bachelor’s Thesis – Initial Presentation
Joseph Birkner - B.Sc. Computational Linguistics / Informatik
LMU/IFI/PMS
Agenda

1. **Vision: Ubiquitous Vertical Search**
2. **Motivation: Encoding of documents**
3. **Objective: A semantic space for documents**
   1. Latent Semantic Spaces
   2. Word2Vec: A semantic space for words
   3. Doc2Vec: A semantic space for documents
4. **Tasks**
   1. Demo
   2. Schedule
5. **References**
Vision: Ubiquitous Vertical Search

• Project IROM
  • “Intelligent Recommendation of Massive Open Online Courses”

• Ubiquitous Vertical Search
  • Every Information Need can be satisfied **instantly** with a vertical search engine
  • Definition of “Vertical Search”: Search within specific domain
  • Solution for specific domain may be applicable to other domains!

• What is Recommendation?
  • “Recommendation” implies Information Need → **Information Retrieval**
  • Information Need is expressed through course query (+ user metadata)
  • The Information Need is satisfied by finding the relevant courses
**Vision:** Ubiquitous Vertical Search

**Domain Information Need**

- User
- Interaction
- Metadata

**Query**

- Click-through data

**Domain Database**

- IMDb

**Domain Database**

- Representation Optimization (Deep Semantic Structured Matching) / DSSM
- Matching Optimization (Deep Relevance Matching Model) / DRMM

**Result**

**Ranked Recommendations**

**NEURAL IR**

**2017-03-30**

Ranking with Neural Network Derived Document Vectors
Agenda

1. **Vision**: Ubiquitous Vertical Search
2. **Motivation**: Encoding of documents
3. **Objective**: A semantic space for documents
   1. Latent Semantic Spaces
   2. Word2Vec: A semantic space for words
   3. Doc2Vec: A semantic space for documents
4. **Tasks**
   1. Prototype
   2. Schedule
5. **References**
Motivation: *Encoding of documents*

- **Axiom**: We need *efficient document representations* to *instantaneously rank* recommended courses based on student need
  - Courses defined by textual descriptions → High Dimensional
- **Kai-Henning Wilker**: “*Multidimensional Clustering of MOOC offers***
  - *Online learning* via SOM/PCA not possible
  - Definition of online learning: New document representations can be generated without expensively re-evaluating all known documents
- **Goal**: Developing an evolving system that can generate good document representations for efficient/effective ranking
- **Good document representation**:
  - Enables *fast* (constant-time) ranking function → Efficiency
  - Ranking seems “*intelligent*” to search engine user → Effectiveness
Motivation: Encoding of documents

- Document Representation in traditional IR: **TF-IDF**
  - (Let \( D \) be a document, \( T_1^N \) all possible Terms, \( \text{tf}(D,T) \) the term frequency function, and \( \text{idf}(T) \) the inverse document freq.)

\[
\text{tf-idf}(D, T_1^N) = \left[ \text{tf}(D, T_i) \ast \text{idf}(D, T_i) \right]_{i=1}^N
\]

- **Problems:**
  - Word order is ignored
  - Flawed word independence assumption

\[
\begin{array}{cccc}
\text{tf-idf} & \text{d[lightsaber]} & \text{d[man]} \\
\hline
\text{D[StarWars]} & 50 & 6 & 2.52 & 126 \\
\text{d[lightsaber]} & 500 & 1e3 & 0.30 & 150 \\
\end{array}
\]

[Introduction Into Programming]

\( D_1 \): This course is a general introduction into programming.

[General History]

\( D_2 \): We programmed this course as a general introduction.

\[ \text{tf-idf}(D_1) = \text{tf-idf}(D_2) = \{ \text{course, general, introduction, program} \} \]
Agenda

1. **Vision:** Ubiquitous Vertical Search
2. **Motivation:** Encoding of documents
3. **Objective:** A *semantic space for documents*
   1. Latent Semantic Spaces
   2. Word2Vec: A semantic space for words
   3. Doc2Vec: A semantic space for documents
4. **Tasks**
   1. Prototype
   2. Schedule
5. **References**
Objective: A semantic space for documents

• **Latent Semantic Spaces (LSS):**
  
  • k-dimensional space $L_k$, where each dimension encodes an orthogonal semantic concept
  
  • Any point $\mathbf{p} \in L_k$ may encode a more complex object $P$
    
    • In this case $\mathbf{p}$ is also called the embedding vector of $P$
  
  • The similarity between two embedding vectors $\mathbf{p}$ and $\mathbf{q}$ can be computed with the cosine function:
    
    • $\text{similarity}(\mathbf{p}, \mathbf{q}) = \cos(\mathbf{p}, \mathbf{q}) = \frac{\mathbf{p} \cdot \mathbf{q}}{|p| * |q|}$
  
  • In the LSS, a ranking may be achieved by embedding both all documents and the user query
Objective: A semantic space for documents

• Example: Word2Vec (Mikolov 2013)
• Traditional representation of Words:
  • One-Hot Term Vector
  • representation(“I like ice cream“) = \[
  \begin{pmatrix}
  1 & 0 & 0 \\
  0 & 0 & 1 \\
  0 & 1 & 0 \\
  \ldots & \ldots & \ldots
  \end{pmatrix}
  \]
  • Typical one-hot encoded word has >10k dimensions \(\rightarrow\) extremely sparse
Objective: A semantic space for documents

• Example: *Word2Vec* (Mikolov 2013)
• Typically ~300 dimensions dense!
• Neural Network also called *Autoencoder*

Input context

Hidden Layer (Encoded word vector)

\[ \sum g(\text{embeddings}) \]

{the, cat, sits, on, the, mat}

Prediction of next word \( \rightarrow \) Backprop.

https://www.tensorflow.org/tutorials/word2vec
Objective: A *semantic space for documents*

- **Doc2Vec**
- Traditional approach:
  - LSA (Latent Semantic Analysis)
  - Reduce tf-idf matrix of all known docs
  - Use Gauss-method to find principal components for low-dimensional Document Vectors.
- **Not an online approach!**
Objective: A semantic space for documents

• **Doc2Vec**
• Creating Document Vectors with RNN
  • RNN: „Recurrent Neural Network“
  • Specifically: Long-Short-Term-Memory/LSTM

1. Use large text corpus to train a large LSTM language model (LM).
  • The corpus may be general or domain-specific (tbd.)
2. Use Language Model to extract Document Feature-Vectors as weight differences after backpropagation.
  • Key idea: Document is characterized by how it differs from „average“.

<table>
<thead>
<tr>
<th>&lt;like&gt;</th>
<th>&lt;ice cream&gt;</th>
</tr>
</thead>
<tbody>
<tr>
<td>[−2.2]</td>
<td>[1.0]</td>
</tr>
<tr>
<td>0</td>
<td>3.6</td>
</tr>
<tr>
<td>0.5</td>
<td>4.3</td>
</tr>
<tr>
<td>4.3</td>
<td>3.0</td>
</tr>
</tbody>
</table>

<table>
<thead>
<tr>
<th>&lt;i&gt;</th>
<th>&lt;like&gt;</th>
<th>&lt;ice cream&gt;</th>
</tr>
</thead>
<tbody>
<tr>
<td>[3e⁻³]</td>
<td>[−2.4]</td>
<td>[1.1]</td>
</tr>
<tr>
<td>0</td>
<td>0.1</td>
<td>3.5</td>
</tr>
<tr>
<td>1.2</td>
<td>1.0</td>
<td>4.2</td>
</tr>
<tr>
<td>2.4</td>
<td>4.5</td>
<td>3.6</td>
</tr>
</tbody>
</table>

http://colah.github.io/posts/2015-08-Understanding-LSTMs
Agenda

1. **Vision:** Ubiquitous Vertical Search
2. **Motivation:** Encoding of documents
3. **Objective:** A semantic space for documents
   1. Latent Semantic Spaces
   2. Word2Vec: A semantic space for words
   3. Doc2Vec: A semantic space for documents
4. **Tasks**
   1. Prototype
   2. Schedule
5. **References**
Tasks: Prototype

- Generated **document vectors** (embeddings)
  - Used the RWTH LM (language model) LSTM (long short-term memory) impl.
- Created 30-dimensional document vectors
  - Only trained with ~1200 course descriptions
  - More dimensions need more training data!
  - Low dimensionality of network might support generalization of language model
Tasks: Schedule

   • Use **different combinations of training data**: Wikipedia abstracts, TREC (Text Retrieval Conference) data, course descriptions.
   • Combine different training sets for **cross-validated hyperparameter search** (architecture, learning rate optimization).

2. Create API to ...
   • ... **generate document/query vectors** from trained LSTMs.
   • ... retrieve ranked set with query among document set.

3. Evaluate ranking performance on TREC datasets.

4. Evaluate select features from the document vectors with heat maps.

5. Bonus: Search for constant shifts in the document space.

[Image of text and diagram]

April

June

July

http://karpathy.github.io/2015/05/21/rnn-effectiveness/

"The unreasonable effectiveness of Recurrent Neural Networks"
References (1)

COHEN, Daniel, AI, Qingyao y CROFT, W Bruce. "Adaptability of neural networks on varying granularity ir

CRASWELL, Nick, CROFT, W Bruce, GUO, Jiafeng, MITRA, Bhaskar y DE RIJKE, Maarten. "Report on the SIGIR
2016 Workshop on Neural Information Retrieval (Neu-IR)".

DEERWESTER, Scott, DUMAIS, Susan T, FURNAS, George W, L, , AUER, Thomas K y HARSCHMAN, Richard.
"Indexing by latent semantic analysis". Journal of the American society for information science. 1990, vol 41,
път. 6, p. 391.

GUO, Jiafeng, FAN, Yixing, AI, Qingyao y CROFT, W Bruce. A deep relevance matching model for ad-hoc

JAECH, Aaron, KAMISSETTY, Hetun, AN, RINGGER, Eric y CLARKE, Charlie. "Match-Tensor: a Deep Relevance

References (2)


