Course Concept Extraction in MOOCs via Embedding-Based Graph Propagation

Reported by: Liangming PAN
Authors: Liangming PAN, Xiaochen WANG, Chengjiang LI,
Juanzi LI and Jie TANG
Knowledge Engineering Group
Tsinghua University

2017-11-28
MOOCs

Massive open online courses (MOOCs) have become increasingly popular and offered students around the world the opportunity to take online courses from prestigious universities.

- World
  - coursera
  - edX
  - Udacity

- China
  - Xuetangx.com
  - Imooc.com
  - 中國大學MOOC
Course concepts refer to the knowledge concepts taught in the course, and the related topics that help students better understand course videos.

You might learn how to write a bubble sort and learn why a bubble sort is not as good as a heapsort. Next, we are going to talk about the quick sort algorithm. Quicksort is an algorithm invented in the 1960s by doctor Tony Hoare. It is also called the partition exchange sort, and is a typical algorithm based on divide-and-conquer.

Now we have the first version of Q sort. After we make an analysis on its performance, performance, we will find that quicksort is an unstable sorting algorithm. Fortunately, the quick sort has an average time complexity of $n \log n$, and in most cases, it can achieve its optimal performance. We first estimate its performance under independent uniform distribution.
Why Course Concept Extraction?

Video-based Structure

Art & Law  Computer Science  Economics

Java Programming  Operating System  Computer Network

JP01  OS01  CN01
JP02  OS02  CN02
JP24  OS30  CN44

Materials  Materials  Materials
Why Course Concept Extraction?

Video-based Structure

Art & Law
Java Programming
JP01
JP02
JP24

Operating System
OS01
OS02
OS30

Computer Science

Economics

Computer Network

Concept-based Structure

Materials

Materials
Why Course Concept Extraction?

- **Motivation 1.** Manually extracting course concepts in MOOCs is infeasible
- **Motivation 2.** A concept map can help improve the learning experience of students
# Related Works: Keyphrase Extraction

<table>
<thead>
<tr>
<th>Category</th>
<th>Method</th>
<th>Authors</th>
<th>Year</th>
</tr>
</thead>
<tbody>
<tr>
<td>Supervised Learning</td>
<td>Naive Bayes</td>
<td>Eibe Frank et al.</td>
<td>IJCAI 1999</td>
</tr>
<tr>
<td></td>
<td>Decision Tree</td>
<td>Peter D. Turney et al.</td>
<td>Journal of IR 2000</td>
</tr>
<tr>
<td></td>
<td>Maximum Entropy</td>
<td>Wentau Yih et al.</td>
<td>WWW 2006</td>
</tr>
<tr>
<td></td>
<td>SVM</td>
<td>Patrice Lopez et al.</td>
<td>2010</td>
</tr>
<tr>
<td>Graph-based Methods</td>
<td>TextRank</td>
<td>Mihalcea R and Tarau P</td>
<td>EMNLP 2004</td>
</tr>
<tr>
<td></td>
<td>ExpandRank</td>
<td>Wan et al.</td>
<td>AAAI 2008</td>
</tr>
<tr>
<td></td>
<td>Topical PageRank</td>
<td>Liu et al.</td>
<td>EMNLP 2010</td>
</tr>
<tr>
<td>Joint Learning-based</td>
<td>Combining Text Summarization and Keyphrase Extraction</td>
<td>Zha et al.</td>
<td>SIGIR 2002</td>
</tr>
<tr>
<td></td>
<td></td>
<td>Wan et al.</td>
<td>ACL 2007</td>
</tr>
</tbody>
</table>
Why Course Concept Extraction *Hard*?

**Low-frequency problem:** Course video captions often contain many course concepts with *low frequency*, primarily for three reasons:

- Course video captions are relatively *short documents*.
- Many infrequent course concepts are from *other prerequisite or related courses*.
- A disambiguated course concept tends to be *expressed in various ways*, which produces many scattered infrequent terms.

You might learn how to write a **bubble sort** and learn why a **bubble sort** is not as good as a **heapsort**. Next, we are going to talk about the **quick sort** algorithm. **Quicksort** is an algorithm invented in the 1960s by doctor Tony Hoare. It is also called the **partition exchange sort**, and is a typical algorithm based on **divide-and-conquer**.

………

Now we have the first version of **Q sort**. After we make an analysis on its performance, we will find that **quicksort** is an **unstable sorting algorithm**. Fortunately, the **quick sort** has an average **time complexity** of \( n \log n \), and in most cases, it can achieve its optimal performance. We first estimate its performance under **independent uniform distribution**.
Properties of Course Concepts

A course concept has the following three properties:

- **Phraseness**
  - A course concept should be a semantically and syntactically correct phrase.

- **Informativeness**
  - A course concept should represent a specific scientific or technical concept.

- **Relatedness**
  - A course concept should be related to a course.

The above properties are hard to be captured by local statistical information because of the Low-frequency problem.
Outline

Backgrounds

Related Works

Methods

Experiments and Analysis

Conclusion
Method Overview

1. Candidate Extraction: Extracting **noun phrases** within K-grams from course video captions based on **linguistic patterns**.
2. Representation: Incorporating **external knowledge** from online encyclopedia to learn **semantic representations** for candidate course concepts.
3. Ranking: Ranking candidate course concepts based on the representation.
**Representation**

- **Phraseness Measurement:** PMI-based method

\[
\begin{align*}
\text{2-grams} & \quad Ph(w_1, w_2) = \frac{2 \times \text{freq}(w_1, w_2)}{\text{freq}(w_1) + \text{freq}(w_2)} \quad (1) \\
\text{N-grams (N>2)} & \quad Ph(t) = \max\{Ph(f_i, b_i) \mid i = 1, \cdots, N - 1\} \quad (2) \\
\text{Averaging} & \quad ph(c) = \alpha \cdot F[ph^D(c)] + (1 - \alpha) \cdot F[ph^E(c)] \quad (3)
\end{align*}
\]
Representation

- Semantic Relatedness

  - Entity Annotation
    - Labeling all entities in Wikipedia Corpus
  
  - Word Embeddings
  
  - Concept Representation
    - Obtaining the vector for each candidate
  
  - Semantic Relatedness
    - Calculating SR by cosine distance

\[
SR(a, b) = \frac{1}{2} \left(1 + \frac{v_a \cdot v_b}{|v_a| \cdot |v_b|}\right)
\]
Course Concept Graph Construction (CCG)

The course concept graph (CCG) of a course is a weighted undirected fully-connected graph denoted as $G = (V,E)$.

- **V** is the *vertex set*: Each vertex in $V$ represents a candidate course concept, associated with a phraseness score.
- **E** is the *edge set*: For an edge $(c_i, c_j) \in E$, its edge weight $e(c_i, c_j) = SR(c_i, c_j)$ indicates the semantic relatedness between $c_i$ and $c_j$, i.e., the likeness of their semantic meaning.

- **Pruning**: An edge $(c_i, c_j)$ exists in a CCG only if $SR(c_i, c_j) > \theta$. 
Ranking

**Assumption:** In CCG, a course concept is likely to connect with other course concepts with high semantic relatedness

**General Idea:** Based on a small *seed set* to find more course concepts in CCG using a *graph-based propagation algorithm*.
Ranking

Propagation Process:

\[
conf^{k+1}(c_i) = \frac{1}{Z} \left( \sum_{c_j \in A(c_i)} \frac{vs^k(c_j, c_i)}{|A(c_i)|} \right)
\]

Voting Score: It determines how much score should a vertex receives from another vertex in each iteration.

\[
vs^k(c_j, c_i) = ph(c_j) \cdot e(c_i, c_j) \cdot conf^k(c_j)
\]

Generalized Voting Score: \(opf(c_i, c_j) = \lambda\) if \(c_i\) and \(c_j\) are overlapping.

\[
gvs^k(c_j, c_i) = opf(c_i, c_j) \cdot ph(c_j) \cdot e(c_i, c_j) \cdot conf^k(c_j)
\]
Experiments

Datasets

<table>
<thead>
<tr>
<th>Dataset</th>
<th>Domain</th>
<th>Language</th>
<th>#courses</th>
<th>#videos</th>
<th>#tokens</th>
<th>#candidates</th>
<th>#labeled</th>
<th>correlation</th>
</tr>
</thead>
<tbody>
<tr>
<td>CSEN</td>
<td>Computer Science</td>
<td>English</td>
<td>8</td>
<td>690</td>
<td>1,242,156</td>
<td>59,050</td>
<td>4,096</td>
<td>0.734</td>
</tr>
<tr>
<td>EcoEN</td>
<td>Economics</td>
<td>English</td>
<td>5</td>
<td>381</td>
<td>401,192</td>
<td>27,571</td>
<td>3,652</td>
<td>0.696</td>
</tr>
<tr>
<td>CSZH</td>
<td>Computer Science</td>
<td>Chinese</td>
<td>18</td>
<td>2,849</td>
<td>2,291,258</td>
<td>79,009</td>
<td>5,309</td>
<td>0.721</td>
</tr>
<tr>
<td>EcoZH</td>
<td>Economics</td>
<td>Chinese</td>
<td>8</td>
<td>455</td>
<td>645,016</td>
<td>60,566</td>
<td>3,663</td>
<td>0.646</td>
</tr>
</tbody>
</table>

Metrics

- R-precision
- MAP (Mean Average Precision)

Baselines

- Statistical-based Methods (TF-IDF, PMI)
- Graph-based Methods (TextRank, Topical PageRank)
Our method outperforms all baselines on all datasets.

TF-IDF & TextRank perform worse than TPR and CCP.

TPR performs better than TextRank across all datasets.

<table>
<thead>
<tr>
<th>Method</th>
<th>CSENM</th>
<th>EcoENM</th>
<th>CSZHM</th>
<th>EcoZHM</th>
</tr>
</thead>
<tbody>
<tr>
<td>TF-IDF Rp</td>
<td>0.125</td>
<td>0.303</td>
<td>0.118</td>
<td>0.198</td>
</tr>
<tr>
<td>MAP</td>
<td>0.105</td>
<td>0.232</td>
<td>0.109</td>
<td>0.145</td>
</tr>
<tr>
<td>PMI Rp</td>
<td>0.239</td>
<td>0.222</td>
<td>0.246</td>
<td>0.179</td>
</tr>
<tr>
<td>MAP</td>
<td>0.141</td>
<td>0.197</td>
<td>0.187</td>
<td>0.121</td>
</tr>
<tr>
<td>TextRank Rp</td>
<td>0.151</td>
<td>0.290</td>
<td>0.142</td>
<td>0.161</td>
</tr>
<tr>
<td>MAP</td>
<td>0.137</td>
<td>0.263</td>
<td>0.131</td>
<td>0.115</td>
</tr>
<tr>
<td>TPR Rp</td>
<td>0.284</td>
<td>0.414</td>
<td>0.305</td>
<td>0.303</td>
</tr>
<tr>
<td>MAP</td>
<td>0.255</td>
<td>0.387</td>
<td>0.267</td>
<td>0.288</td>
</tr>
<tr>
<td>CCP Rp</td>
<td>0.443</td>
<td>0.427</td>
<td>0.434</td>
<td>0.435</td>
</tr>
<tr>
<td>MAP</td>
<td>0.432</td>
<td>0.365</td>
<td>0.416</td>
<td>0.423</td>
</tr>
</tbody>
</table>
Outline

- Backgrounds
- Related Works
- Methods
- Experiments and Analysis
- Conclusion
Conclusion

- Automatically discovering course concepts in MOOCs

Future Directions

- Research on automatically course concept map generation
- Try deep learning models for course concept extraction
- Incorporating dynamic information in MOOCs (e.g., user behavior, forums, QA between students and teachers).
Thanks!

Liangming Pan
KEG, THU
peterpan10211020@163.com