Machine Learning for Improving Search Relevance

Jun Xu
Microsoft Research Asia
Short Biography

• Jun Xu
  – PHD at Nankai University, China (2006)
  – Associate Researcher at Microsoft Research Asia (2006 ~ today)

• Research: improving web search using machine learning techniques
Short Biography (cont')

Jun Xu

I am an associate researcher at Microsoft Research Asia (MSRA), Information Retrieval and Mining (IRM) Group.

I obtained a B.S. in July 2001 and a Ph.D. in Computer Application and Technology in July 2006, both from Nankai University. My advisor is professor Huang Ye-lou. Thesis: Cost-sensitive Learning of Ranking for Information Retrieval.

I participated in the Microsoft Research Asia Internship Program from September 2003 to December 2005 as a member of Natural Language Computing Group. My mentor is Dr. Hang Li.

My major research interest includes text mining, machine learning, and web search.

I have published several papers in the fields of information retrieval and machine learning. My work has been cited over 200 times, as shown in the citation graph. My h-index is 10, indicating that I have published 10 papers that have each been cited at least 10 times.

Contact Information
Microsoft Research Asia,
13F, Microsoft Building 2, No. 5 Danling Street, Haidian Distinct
Beijing, China 100080
Email: junxu AT microsoft.com
People Who Also Contributed to This Talk

Hang LI

Wei WU

Quan WANG
Outline

1. Matching between Query and Document
2. Matching with Translation Model
3. Matching with Topic Model
4. Summary and Open Problems
1. Matching between Query and Document
A Good Web Search Engine

• Must be good at
  – Relevance
  – Freshness
  – Comprehensiveness
  – User interface

• Relevance is particularly important
Query Document Mismatch is Biggest Challenge in Web Search
Same Search Intent Different Query Representations
Example = “Distance between Sun and Earth”

- "how far" earth sun
- "how far" sun
- "how far" sun earth
- average distance earth sun
- average distance from earth to sun
- average distance from the earth to the sun
- distance between earth & sun
- distance between earth and sun
- distance between earth and the sun
- distance from earth to the sun
- distance from sun to earth
- distance from sun to the earth
- distance from the earth to the sun
- distance from the sun to earth
- distance from the sun to the earth
- distance of earth from sun
- distance between earth sun
- how far away is the sun from earth
- how far away is the sun from the earth
- how far earth from sun
- how far earth from the sun
- how far from earth is the sun
- how far from earth to sun
- how far from the earth to the sun
- distance between sun and earth
- distance between sun and earth
**Same Search Intent, Different Query Representations**

Example = “Youtube”

<table>
<thead>
<tr>
<th>Query</th>
<th>Query</th>
<th>Query</th>
</tr>
</thead>
<tbody>
<tr>
<td>yutube</td>
<td>yuotube</td>
<td>yuo tube</td>
</tr>
<tr>
<td>ytube</td>
<td>youtubr</td>
<td>yu tube</td>
</tr>
<tr>
<td>youtubo</td>
<td>youtuber</td>
<td>youtubecom</td>
</tr>
<tr>
<td>youtube om</td>
<td>youtube music videos</td>
<td>youtube videos</td>
</tr>
<tr>
<td>youtube</td>
<td>youtube com</td>
<td>youtube co</td>
</tr>
<tr>
<td>youtub com</td>
<td>you tube music videos</td>
<td>yout tube</td>
</tr>
<tr>
<td>youtub</td>
<td>you tube com yourtube</td>
<td>your tube</td>
</tr>
<tr>
<td>you tube</td>
<td>you tub</td>
<td>you tube video clips</td>
</tr>
<tr>
<td>you tube videos</td>
<td>www you tube com</td>
<td>wwwwww youtube com</td>
</tr>
<tr>
<td>www youtube</td>
<td>www youtube com</td>
<td>www youtube co</td>
</tr>
<tr>
<td>yotube</td>
<td>www you tube</td>
<td>www u tube com</td>
</tr>
<tr>
<td>ww youtube com</td>
<td>www utube</td>
<td>www u tube</td>
</tr>
<tr>
<td>utube videos</td>
<td>utube com</td>
<td>utube</td>
</tr>
<tr>
<td>u tube com</td>
<td>utub</td>
<td>u tube videos</td>
</tr>
<tr>
<td>u tube</td>
<td>my tube</td>
<td>toutube</td>
</tr>
<tr>
<td>outube</td>
<td>our tube</td>
<td>toutube</td>
</tr>
</tbody>
</table>
Matching between Two Worlds:
In Principle, Language Understanding Is Needed
Challenges in Machine Learning for Matching

- How to leverage relations in data and prior knowledge
- Scale is very large
Previous Work

• Studied in long history of IR
• Query expansion, pseudo relevance feedback
• Latent Semantic Indexing, Probabilistic Latent Semantic Indexing
• ... ...
New Trends in Recent Work

- Employing more machine learning (supervised and unsupervised)
- Large scale
- Use of log data

- This talk focuses on recent work!
Approaches to Learning for Matching Between Query and Document

• Matching with Translation Model
• Matching with Topic Model
• Matching by Query Reformulation
• Matching with Dependency Model
• Matching in Latent Space
2. Matching with Statistical Machine Translation
Outline of Section 2

• Statistical Machine Translation
• Matching with Translation Model
• Issues in Matching with Translation Model
• Methods for Matching with Translation Models
Statistical Machine Translation (SMT)

• Given sentence $C$ in source language, translates it into sentence $E$ in target language

$$E^* = \arg\max_E P(E|C)$$

• Linear combination of features

$$P(E|C) = \frac{1}{Z(C, E)} \exp \sum_i \lambda_i h_i(C, E)$$

$$E^* = \arg\max_E \sum_i \lambda_i h_i(C, E)$$
Typical Translation Models

• Word-based
  – Translating word to word

• Phrase-based
  – Translating based on phrase

• Syntax-based
  – Translating based on syntactic structure
IBM Model One
(Brown et al., 1993)

- Generating target sentence
  - Length $M$ of target sentence is generated
  - For each target sentence position, $i = 1: M$
    - Word $c_j$ in source sentence $C$ is selected
    - $e_i$ at position $i$ is generated depend on $c_j$

$$P(E|C) = \epsilon \prod_{i=1}^{M} \sum_{j=1}^{N} P(e_i|c_j)$$
Matching with Translation Model

• Translating document $d$ to query $q$ (or translation document language model to query language model)

• Given query $q$ and document $d$, translation probability is viewed as matching score between $q$ and $d$

• Difference from conventional translation model
  – Translation in same language
  – Self translation plays important role
Addressing Term Mismatch with Translation Model

- Translation probability $P(q|w)$ represents matching degree between words in query and document

| $q$       | $P(q|w)$ | $q$       | $P(q|w)$ |
|-----------|----------|-----------|----------|
| titanic   | 0.56218  | Vista     | 0.80575  |
| ship      | 0.01383  | Windows   | 0.05344  |
| movie     | 0.01222  | Download  | 0.00728  |
| pictures  | 0.01211  | ultimate  | 0.00571  |
| sink      | 0.00697  | xp        | 0.00355  |
| facts     | 0.00689  | microsoft | 0.00342  |
| photos    | 0.00533  | bit       | 0.00286  |
| rose      | 0.00447  | compatible| 0.00270  |
| people    | 0.00441  | premium   | 0.00244  |
| survivors | 0.00369  | free      | 0.00211  |

$w = \text{titanic}$ $w = \text{vista}$
Approaches to Matching with Translation Model

• Translating document to query

\[ P(q|d) \]

• Translating document model to query model

Matching based on query language model
Issues in Matching with Translation Models

• Types of Training Data
• Types of Document Fields
• Types of Translation Models
Types of Training Data for Learning Translation Probabilities

- Synthetic data (Berger & Lafferty, ’99)
- Title-body pairs of documents (Jin et al., ’02)
- Query-title pairs in click-through data (Gao et al., ’10)

http://webmessenger.msn.com
title: “msn web messenger”

<table>
<thead>
<tr>
<th>clicked queries</th>
<th>score</th>
</tr>
</thead>
<tbody>
<tr>
<td>msn web</td>
<td>0.6675</td>
</tr>
<tr>
<td>webmenssenger</td>
<td>0.6621</td>
</tr>
<tr>
<td>msn online</td>
<td>0.6403</td>
</tr>
<tr>
<td>Windows web messanger</td>
<td>0.6321</td>
</tr>
<tr>
<td>talking to friends on msn</td>
<td>0.6130</td>
</tr>
<tr>
<td>... ...</td>
<td>... ...</td>
</tr>
</tbody>
</table>
Types of Document Fields

- Use of title is better than body (Huang et al., ‘10)
- Titles and queries have similar languages
- Bodies and queries have very different languages

<table>
<thead>
<tr>
<th>Order</th>
<th>Body</th>
<th>Anchor</th>
<th>Title</th>
<th>Query</th>
</tr>
</thead>
<tbody>
<tr>
<td>Unigram</td>
<td>13242</td>
<td>4164</td>
<td>3633</td>
<td>1754</td>
</tr>
<tr>
<td>Bigram</td>
<td>5567</td>
<td>966</td>
<td>1420</td>
<td>289</td>
</tr>
<tr>
<td>Trigram</td>
<td>5381</td>
<td>740</td>
<td>1299</td>
<td>180</td>
</tr>
<tr>
<td>4-gram</td>
<td>5785</td>
<td>731</td>
<td>1382</td>
<td>168</td>
</tr>
</tbody>
</table>
Methods for Matching with Translation Models

• Translating document to query
  – Word-based model (Berger & Lafferty, ’99)
  – Phrase-based model (Gao et al., ’10)
  – Syntax-based model (Park et al., ’11)
  – Topic-based model (Gao et al., ’11)
  – Learning translation probabilities from documents (Karimzadehgan & Zhai, ’10)

• Translating document model to query model
  – Translated query language model (Jin et al., ’02)
Matching with Word-based Translation Model

• Basic model

\[ P(q|d) = \prod_{q \in q} P(q|d) = \prod_{q \in q} \sum_{w \in d} P(q|w)P(w|d) \]

- translation probability
- document language model

• Smoothing to avoid zero translation probability (Berger & Lafferty, ’99)

\[ P(q|d) = \prod_{q \in q} \left( \alpha P(q|coll) + (1 - \alpha) \sum_{w \in d} P(q|w)P(w|d) \right) \]

- background unigram model

• Adding self-translation (Gao et al., ’10)

\[ P(q|d) = \prod_{q \in q} \left( \alpha P(q|coll) + (1 - \alpha) \left( \beta P(q|d) + (1 - \beta) \sum_{w \in d} P(q|w)P(w|d) \right) \right) \]

- unsmoothed document model
Examples of Translation Probabilities

| q            | t(q | w) | q            | t(q | w) | q            | t(q | w) |
|--------------|------|--------------|------|--------------|------|
| solzhenitsyn | 0.319| carcinogen   | 0.667| zubin_mehta  | 0.248|
| citizenship  | 0.049| cancer       | 0.032| zubin        | 0.139|
| exile        | 0.044| scientific   | 0.024| mehta        | 0.134|
| archipelago  | 0.030| science      | 0.014| philharmonic | 0.103|
| alexander    | 0.025| environment  | 0.013| orchestra    | 0.046|
| soviet       | 0.023| chemical     | 0.012| music        | 0.036|
| union        | 0.018| exposure      | 0.012| bernstein    | 0.029|
| komsomolskaya| 0.017| pesticide     | 0.010| york         | 0.026|
| treason      | 0.015| agent         | 0.009| end          | 0.018|
| vishnevskaya | 0.015| protect       | 0.008| sir          | 0.016|

w = solzhenitsyn  

| q            | t(q | w) | q            | t(q | w) | q            | t(q | w) |
|--------------|------|--------------|------|--------------|------|
| pontiff      | 0.502| everest      | 0.439| wildlife     | 0.705|
| pope         | 0.169| climb        | 0.057| fish         | 0.038|
| paul         | 0.065| climber      | 0.045| acre         | 0.012|
| john         | 0.035| whitaker     | 0.039| species      | 0.010|
| vatican      | 0.033| expedition   | 0.036| forest       | 0.010|
| ii           | 0.028| float        | 0.024| environment  | 0.009|
| visit        | 0.017| mountain     | 0.024| habitat      | 0.008|
| papal        | 0.010| summit       | 0.021| endangered   | 0.007|
| church       | 0.005| highest      | 0.018| protected    | 0.007|
| flight       | 0.004| reach        | 0.015| bird         | 0.007|

w = pontiff  

w = everest  

w = wildlife
Matching with Phrase-based and Syntax-based Translation Models

- Phrase-based translation model (Gao et al., ’10)
  
  \[
P(q|d) \approx \sum_{(S,T,M) \in B(q,d)} P(T|d,S)P(M|d,S,T)
\]

- Syntax-based translation model (Park et al., ’11)
  - Queries and documents are parsed to syntax trees
  - Translation probabilities calculated based on parsed trees
Topic-based Translation Model  
(Gao et al., 2011)

• Query and document use different vocabularies to express the same distribution of topics

\[ P(q|d) = \prod_{q \in q} P_{bitm}(q|d) = \prod_{q \in q} \sum_z P(q|\phi^q_z) P(z|\theta^d) \]

• Smoothing and addressing self translation

\[ P_s(q|d) = \prod_{q \in q} (\lambda_1 P(q|C) + (1 - \lambda_1)(\lambda_2 P(q|d) + (1 - \lambda_2)P_{bitm}(q|d))) \]
Learning Translation Probabilities from Documents (Karimzadehghan & Zhai, ’10)

- **Mutual information of words** \((w, u)\)

\[I(w; u) = \sum_{x_w=0,1} \sum_{x_u=0,1} p(X_w, X_u) \log \frac{p(X_w, X_u)}{p(X_w)p(X_u)}\]

- **Translation probability**

\[P_t(w|u) = \begin{cases} 
(1 - \alpha) \frac{I(w; u)}{\sum_{w'} I(w'; u)} & \text{if } w \neq u \\
\alpha + (1 - \alpha) \frac{I(u; u)}{\sum_{w'} I(w'; u)} & \text{if } w = u 
\end{cases}\]
Matching with Translated Query Language Model

(Jin et al., ’02)

\[ P(q|d, M) = \epsilon \prod_{q_i \in q} \lambda \left( \frac{P(q_i|\phi, M)}{|d| + 1} \right) + \sum_{w \in d} P(q_i|w, M)P(w|d) + (1 - \lambda)P(q_i|GE) \]

- translate doc word to query word
- document language model
- background language model
References

References

3. Matching with Topic Model
Outline of Section 3

• Topic Modeling
• Methods of Matching with Topic Model
• Two Approaches to Topic Modeling
In machine learning and natural language processing, a topic model is a type of statistical model for discovering the abstract "topics" that occur in a collection of documents.

- Wikipedia

... algorithms to uncover the hidden thematic structure of a collection of documents. These algorithms help us develop new ways to search, browse and summarize large archives of texts.

- David M. Blei

topic = a group of words with weights

<table>
<thead>
<tr>
<th>Topic1</th>
<th>OPEC</th>
<th>oil</th>
<th>cent</th>
<th>barrel</th>
<th>price</th>
</tr>
</thead>
<tbody>
<tr>
<td>Topic2</td>
<td>Africa</td>
<td>South</td>
<td>African</td>
<td>Angola</td>
<td>Apartheid</td>
</tr>
</tbody>
</table>
Topic Model (cont’)

- **Input**
  - Document collection
- **Processing**
  - Discover latent topics in document collection
- **Output**
  - Latent topics in document collection
  - Topic representations of documents
Topic Model: Two Approaches

• Probabilistic approach

\[ d \rightarrow z \rightarrow w \]

\[ M \times N \]

• Non-probabilistic approach

\[ D \approx U \times V^T \]
Deal with Term Mismatch with Topic Model

- Topics of query and document are identified
- Match query and document through topics, although query and document do not share terms

<table>
<thead>
<tr>
<th>Topic1</th>
<th>Topic2</th>
<th>Topic3</th>
<th>Topic4</th>
<th>Topic5</th>
<th>Topic6</th>
<th>Topic7</th>
<th>Topic8</th>
<th>Topic9</th>
<th>Topic10</th>
</tr>
</thead>
<tbody>
<tr>
<td>OPEC</td>
<td>Africa</td>
<td>contra</td>
<td>school</td>
<td>Noriega</td>
<td>firefight</td>
<td>plane</td>
<td>Saturday</td>
<td>Iran</td>
<td>senate</td>
</tr>
<tr>
<td>oil</td>
<td>South</td>
<td>Sandinista</td>
<td>student</td>
<td>Panama</td>
<td>ACR</td>
<td>crash</td>
<td>coastal</td>
<td>Iranian</td>
<td>Reagan</td>
</tr>
<tr>
<td>cent</td>
<td>African</td>
<td>rebel</td>
<td>teacher</td>
<td>Panamanian</td>
<td>forest</td>
<td>flight</td>
<td>estimate</td>
<td>Iraq</td>
<td>billion</td>
</tr>
<tr>
<td>barrel</td>
<td>Angola</td>
<td>Nicaragua</td>
<td>education</td>
<td>Delval</td>
<td>park</td>
<td>air</td>
<td>western</td>
<td>hostage</td>
<td>budget</td>
</tr>
<tr>
<td>price</td>
<td>apartheid</td>
<td>Nicaraguan</td>
<td>college</td>
<td>canal</td>
<td>blaze</td>
<td>airline</td>
<td>Minsch</td>
<td>Iraqi</td>
<td>Trade</td>
</tr>
</tbody>
</table>
Methods of Matching Using Topic Model

• Topic level matching
  – Probabilistic model: PLSI (Hofmann ’99), LDA (Blei et al., ’03)
  – Non-probabilistic model: LSI (Deerwester et al., ’88), NMF (Lee & Seung ’00), RLSI (Wang et al., ’11)

• Document smoothing
  – Clustering-based (Kurland & Lee ’04, Diaz ’05)
  – LDA-based (Wei & Croft ’06)

• Query smoothing
  – PLSI-based (Yi & Allan ’09)
Topic Level Matching

• Representing query and document as topic distributions (or topic vectors)
  – \( q \rightarrow P(z|q) \)
  – \( d \rightarrow P(z|d) \)

• Matching between topic distributions (or topic vectors)
  – Cosine similarity
  – Symmetric KL-divergence:
    \[
    \sum_z \left( P(z|q) \ln \frac{P(z|q)}{P(z|d)} \right) + \sum_z \left( P(z|d) \ln \frac{P(z|d)}{P(z|q)} \right)
    \]
Document Smoothing with Topics
(Wei & Croft, 2006)

• Topic model: PLSI

\[ P_{PLSI}(w|d) = \sum_z P(w|z)P_{PLSI}(z|d) \]

• Topic model: LDA

\[ P_{LDA}(w|d) = \sum_z P(w|z)P_{LDA}(z|d) \]

• Combination of language model and topic model

\[ P(w|d) = \alpha P_{LM}(w|d) + (1 - \alpha)P_{TM}(w|d) \]
Query Smoothing with Topic Model (Yi & Allan, 2009)

• Topic model

\[ P_{TM}(w|q) = \sum_z P(w|z)P(z|q) \]

• Generate words from topic model

• Query expansion with generated words
Two Approaches to Topic Modeling

• Probabilistic approach
  – Model: probabilistic model (graphical model)
  – Learning: maximum likelihood estimation
  – Methods: PLSI, LDA

• Non-probabilistic approach
  – Model: vector space model
  – Learning: matrix factorization
  – Methods: LSI, NMF, RLSI

• Non-probabilistic models can be reformulated as probabilistic models
Probabilistic Topic Model

• Topic: probability distribution over words
• Document: probability distribution over topics
• Graphical model
  – Word, topic, document, and topic distribution are represented as nodes
  – Probabilistic dependencies are represented as directed edges
  – Generation process
• Interpretation: soft clustering
Probabilistic Latent Semantic Indexing (Hofmann 1999)

• For each document
  – Generate doc $d$ with probability $P(d)$
  – For each word
    • Generate topic $z$ with probability $P(z|d)$
    • Generate word $w$ with probability $P(w|z)$
Latent Dirichlet Allocation
(Blei et al., 2003)

• Generation process
  – Word distribution given topic \( \phi \sim \text{Dir}(\beta) \)
  – For each document:
    • Determine topic distribution \( \theta \sim \text{Dir}(\alpha) \)
    • For each word:
      – Generate topic \( z \sim \text{Mul}(\theta) \)
      – Generate word \( w \sim \text{Mul}(\phi) \)
Non-probabilistic Topic Model

- Document: vector of words
- Topic: vector of words
- Document representation: combination of topic vectors
- Matrix factorization
- Interpretation: projection to topic space
Latent Semantic Indexing (Deerwester et al., 1990)

- Representing document collection with co-occurrence matrix (TF or TFIDF)
- Performing Singular Value Decomposition (SVD) and producing k-dimensional topic space
Nonnegative Matrix Factorization
(Lee and Seung, 2001)

- $\mathbf{U}$ and $\mathbf{V}$ are nonnegative

$$\min_{\mathbf{U}, \mathbf{V}} \| \mathbf{D} - \mathbf{U}\mathbf{V}^T \|_F$$

$$s.t. u_{ij} \geq 0; v_{ij} \geq 0$$
Regularized Latent Semantic Indexing  
(Wang et al., 2011)

• Topics are sparse

\[
\min_{U,V} \sum_{n=1}^{N} \left\| d_n - Uv_n \right\|_2^2 + \lambda_1 \sum_{k=1}^{K} \left\| u_k \right\|_1 + \lambda_2 \sum_{n=1}^{N} \left\| v_n \right\|_2^2
\]

topics are sparse
Probabilistic Interpretation of Regularized Latent Semantic Indexing

- Document generated according to Gaussian distribution
  \[ P(\mathbf{d}_n | \mathbf{U}, \mathbf{v}_n) \propto \exp(-\|\mathbf{d}_n - \mathbf{Uv}_n\|_2^2) \]
- Laplacian prior
  \[ P(\mathbf{u}_k) \propto \exp(-\lambda_1 \|\mathbf{u}_k\|_1) \]
- Gaussian prior
  \[ P(\mathbf{v}_n) \propto \exp(-\lambda_2 \|\mathbf{v}_n\|_2^2) \]
References

4. Summary and Open Problems
Summary of Talk

• Query document matching is biggest challenge in search
• Machine learning for matching between query and document is making progress
• Approaches to Learning for Matching Between Query and Document
  – Statistical machine translation
  – topic modeling
  – ...
  – ...
Challenges and Open Problems

• Evaluation measures
  – Cranefield approach has limitation

• Topic drift
  – Language is synonymous and polysemous

• Scalability
  – E.g., topic modeling needs large scale computing environment
Thank You!

junxu@microsoft.com
http://research.microsoft.com/~junxu