Matching Reviews
to Objects using a Language Model

Nilesh Dalvi, Ravi Kumar, Bo Pang, Andrew Tomkins
Yahoo! Research
ACL and AFNLP, 2009

15 April, 2014
Jaehwan Lee
Outline

- Introduction
- Related Work
- Model and Method
- Data
- Evaluation
- Conclusions
Introduction

- the Search Engine would like
  - to offer a high quality result set for even obscure restaurants
  - to enable advanced applications and recommendation

- To solve them, It faces two high-level challenges
  - identify the restaurant review pages on the Web
  - identify the restaurant that is being reviewed

- Notice
  - restaurant reviews are running example
  - “the techniques are general”
Introduction

- the Search Engine would like
  - to offer a high quality result set for even obscure restaurants
  - to enable advanced applications and recommendation

- To solve them, It faces two high-level challenges
  - identify the restaurant review pages on the Web
  - identify the restaurant that is being reviewed

- Notice
  - restaurant reviews are running example
  - “the techniques are general”
Introduction

Two Settings of Related Flavor

- Entity Matching
  - to find the correspondence between two structured objects

- Information Retrieval (IR)
  - to match unstructured short text against unstructured text
Introduction
Classical IR Methods Doesn’t Fit

- **Example of “Food”**
  - “food” is rare as a restaurant name
  - thus, it will get a very high IDF score
  - AND hence will likely be the top match for all reviews containing the word “food”

- **UNLIKE in traditional IR**
  - a query (i.e. review) is long and a document (i.e. restaurant) is short
The intuition behind their model is simple and natural

- When a review is written about an object,
- each word in the review is drawn either from a description of the object or from a generic review language that is independent of the object.
Related Work

- **Opinion topic identification**
  - Some work on fine-grained opinion extraction from reviews
  - focused on identifying product features of the object under review, rather than object itself

- **Language modeling**
  - to postulate a model for each document
  - to select the document that is most likely to have generated for a given query

- **Entity matching**
  - consider pairwise attribute similarities between entities
  - exploit the relationships that exist between entities
Model and Method

- r: a review
- $\mathcal{R}$: a collection of reviews
- e: an object, has a set of attributes
- $\mathcal{E}$: a set of objects
- $\text{text}(e)$: the union of the textual content of all its attributes
- $r_e = r \cap \text{text}(e)$

- $P(w)$: the probability the word $w$ is chosen according some object-independent distribution
- $P_e(w)$: the probability the word $w$ is chosen according some object-dependent distribution
Model and Method

Review Language Model (RLM)

\[ \Pr[r | e] = Z(r) \prod_{w \in r} \Pr[w | e] \]

\[ = Z(r) \prod_{w \in r} ((1 - \alpha)P(w) + \alpha P_e(w)), \quad (1) \]

- It represents the probability that a review \( r \) is a review about object \( e \) when \( e \) exists in \( r \)
- \( \alpha \) is a parameter \( (0 < \alpha < 1) \)
- Modeling
  - \( P_e(w) \) is object-dependent
  - \( P(w) \) is object-independent (generic review feature)
Review Language Model (RLM)

\[
\Pr[r \mid e] = Z(r) \prod_{w \in r} \Pr[w \mid e]
= Z(r) \prod_{w \in r} ((1 - \alpha) P(w) + \alpha P_e(w)), \quad (1)
\]

It can be zero, if a word \( w \) is not in text(e)
Thus, have to modify the equation as following

\[
\Pr[r \mid e] = Z(r) \prod_{w \in r \setminus r_e} (1 - \alpha) P(w) \cdot \prod_{w \in r_e} ((1 - \alpha) P(w) + \alpha P_e(w))
= Z(r) \prod_{w \in r} (1 - \alpha) P(w) \cdot \prod_{w \in r_e} \left(1 + \frac{\alpha}{1 - \alpha} \frac{P_e(w)}{P(w)}\right). \quad (2)
\]
Model and Method

Review Language Model (RLM)

By assuming a uniform distribution for \( \Pr[e] \), we get

\[
e^* = \arg \max_e \Pr[e | r] = \arg \max_e \frac{\Pr[e]}{\Pr[r]} \cdot \Pr[r | e].
\]

By assuming a uniform distribution for \( \Pr[e] \), we get

\[
e^* = \arg \max_e \Pr[r | e],
\]

\[
e^* = \arg \max_e \log \Pr[r | e].
\]

Since \( Z(r) \prod_{w \in r} ((1 - \alpha) P(w)) \) is independent of \( e \), using (2), we have

\[
e^* = \arg \max_e \sum_{w \in r_e} \log \left( 1 + \frac{\alpha}{1 - \alpha} \frac{P_e(w)}{P(w)} \right).
\]

(3)
Model and Method

Review Language Model (RLM)

By assuming a uniform distribution for \( \Pr[e] \), we get

\[
e^* = \arg \max_e \Pr[e \mid r] = \arg \max_e \frac{\Pr[e]}{\Pr[r]} \cdot \Pr[r \mid e].
\]

By assuming a uniform distribution for \( \Pr[e] \), we get

\[
e^* = \arg \max_e \Pr[r \mid e],
\]

\[
e^* = \arg \max_e \log \Pr[r \mid e].
\]

Since \( Z(r) \prod_{w \in r} ((1 - \alpha)P(w)) \) is independent of \( e \), using (2), we have

\[
e^* = \arg \max_e \sum_{w \in r} \log \left(1 + \alpha \frac{P_e(w)}{1 - \alpha P(w)}\right).
\] (3)
Model and Method

Review Language Model (RLM)

- **Object-independent factor**

  \[
  P(w) = \frac{c(w, R(g)) + 1}{\sum_{w'} c(w', R(g)) + |V|},
  \]

  - By treating the set of processed reviews where for each review-object pair \((r, e)\), words in text \(\mathcal{E}\) are remove from \(r\) as an approximation of \(R(g)\).
  - Then, we can compute \(P(w)\) in the aforementioned manner.

- **Object-dependent factor**

  \[
  (\text{say, } g(w) = \log(1/f_w)), \text{ we let } \]
  \[
  P_e(w) = \frac{g(w)}{\sum_{w' \in \text{text}(e)} g(w')} .
  \]

  - By using the frequency \(f_w\) of the word \(w\) in \(R\) or in \(\{\text{text}(e) | e \in \mathcal{E}\}\).
Model and Method

RLM, TFIDF and TFIDF+

- Generic equation

\[
e^* = \arg \max_e \sum_{w \in \tau_e} \log f(w),
\]

- for RLM, \( f(w) \) goes

\[
f(w) = f_R(w) = 1 + \frac{\alpha}{1 - \alpha} \frac{P_e(w)}{P(w)}
\]

- for TFIDF and TFIDF+, \( f(w) \) goes

\[
f(w) = f_B(w) = \frac{1}{Q(w)}
\]

\[
Q(w) = \frac{df(w)}{N}
\]
Data

- 299,762 reviews
  - each aligned with one of a set of 12,408 unique restaurants hosted on Yelp (yelp.com)
  - no more than 40 reviews per each restaurant

- 681,320 restaurants from Yahoo! Local database

Task
- to match a given Yelp review, using ONLY its free-form textual content
Data

The Final Aligned Dataset

- $R$
  - 24,910 Yelp reviews covering 6,010 restaurants
- $R'$
  - to estimate the models
  - reviews filtered out because of lack of identifying information were added
  - 205,447 reviews
- $R_{test}$
  - to evaluate RLM
  - 11,217 reviews

- There are no overlapping restaurants between them
Evaluation

- Unlike a standard IR task
  - not interested in retrieving multiple relevant objects
  - each review in dataset has only one single correct match from $E$

- Macro vs. micro average
  - Macro average
    - first, compute the average for reviews about the same restaurant
    - and report the average over all restaurants
  - micro average
    - take the average accuracy over all reviews

- Accuracy @ k
  - consider a review is correctly matched if one of the top-$k$ objects returned is the correct match
### Evaluation

#### Main Result

<table>
<thead>
<tr>
<th></th>
<th></th>
<th></th>
</tr>
</thead>
<tbody>
<tr>
<td>RLM</td>
<td>0.647</td>
<td>0.576</td>
</tr>
<tr>
<td>TFIDF⁺</td>
<td>0.518</td>
<td>0.481</td>
</tr>
<tr>
<td>TFIDF</td>
<td>0.314</td>
<td>0.317</td>
</tr>
</tbody>
</table>

(a) Main comparison.

<table>
<thead>
<tr>
<th></th>
<th></th>
<th></th>
</tr>
</thead>
<tbody>
<tr>
<td>RLM-UNIFORM</td>
<td>0.634</td>
<td>0.562</td>
</tr>
<tr>
<td>RLM-UNCUT</td>
<td>0.627</td>
<td>0.546</td>
</tr>
<tr>
<td>RLM-DECAP</td>
<td>0.640</td>
<td>0.573</td>
</tr>
</tbody>
</table>

(b) RLM variants.

<table>
<thead>
<tr>
<th></th>
<th></th>
<th></th>
</tr>
</thead>
<tbody>
<tr>
<td>TFIDF⁺-N</td>
<td>0.586</td>
<td>0.523</td>
</tr>
<tr>
<td>TFIDF⁺-D</td>
<td>0.593</td>
<td>0.533</td>
</tr>
<tr>
<td>TFIDF⁺-O</td>
<td>0.522</td>
<td>0.488</td>
</tr>
<tr>
<td>TFIDF⁺-ND</td>
<td>0.628</td>
<td>0.549</td>
</tr>
<tr>
<td>TFIDF⁺-NDO</td>
<td>0.647</td>
<td>0.576</td>
</tr>
</tbody>
</table>

(c) TFIDF⁺ variants.

Table 1: Average accuracy of the top-1 prediction for various techniques. Micro-average computed over 11,217 reviews in $R_{test}$; macro-average computed over 2,810 unique restaurants in $R_{test}$. 
Evaluation

Main Result

Figure 1: Precision–recall curve (of top one prediction): RLM vs. TFIDF+ baseline.

Figure 2: Accuracy@k (percentage of reviews whose correct match is returned in one of its top-k predictions): RLM vs. TFIDF+ baseline.
Evaluation

Main Result

Longer reviews might be more difficult to match since they may include more proper nouns such as dish names and related restaurants, and yield a longer list of highly competitive candidate objects.

Figure 3: Average accuracy of the top-1 prediction for reviews with different length (on test set): RLM vs. TFIDF⁺ baseline.
Evaluation

Main Result

- **Choices for RLM**
  - RLM-Uniform
  - RLM-Uncut
  - RLM-Decap

- **Revisiting TFIDF+**
  - Object Length Normalization
  - Dampening
  - Removing mentions of objects

- **Using term counts**
  - each of the other modeling decisions incorporated in RLM is important

Table 1: Average accuracy of the top-1 prediction for various techniques. Micro-average computed over 11,217 reviews in $R_{test}$; macro-average computed over 2,810 unique restaurants in $R_{test}$. 

<table>
<thead>
<tr>
<th></th>
<th></th>
<th></th>
</tr>
</thead>
<tbody>
<tr>
<td>RLM</td>
<td>0.647</td>
<td>0.576</td>
</tr>
<tr>
<td>TFIDF</td>
<td>0.518</td>
<td>0.481</td>
</tr>
<tr>
<td>TFIDF</td>
<td>0.314</td>
<td>0.317</td>
</tr>
</tbody>
</table>

(a) Main comparison.

<table>
<thead>
<tr>
<th></th>
<th></th>
<th></th>
</tr>
</thead>
<tbody>
<tr>
<td>RLM-UNIFORM</td>
<td>0.634</td>
<td>0.562</td>
</tr>
<tr>
<td>RLM-UNCUT</td>
<td>0.627</td>
<td>0.546</td>
</tr>
<tr>
<td>RLM-DECAP</td>
<td>0.640</td>
<td>0.573</td>
</tr>
</tbody>
</table>

(b) RLM variants.

<table>
<thead>
<tr>
<th></th>
<th></th>
<th></th>
</tr>
</thead>
<tbody>
<tr>
<td>TFIDF+ -N</td>
<td>0.586</td>
<td>0.523</td>
</tr>
<tr>
<td>TFIDF+ -D</td>
<td>0.593</td>
<td>0.533</td>
</tr>
<tr>
<td>TFIDF+ -O</td>
<td>0.522</td>
<td>0.488</td>
</tr>
<tr>
<td>TFIDF+ -ND</td>
<td>0.628</td>
<td>0.549</td>
</tr>
<tr>
<td>TFIDF+ -NDO</td>
<td>0.647</td>
<td>0.576</td>
</tr>
</tbody>
</table>

(c) TFIDF+ variants.
Conclusions

- The model provides us a principled way to match reviews to objects

- Their techniques vastly outperforms standard TF-IDF based techniques