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
  - <u>identify the restaurant that is being reviewed</u>
- Notice
  - restaurant reviews are running example
  - "the techniques are general"

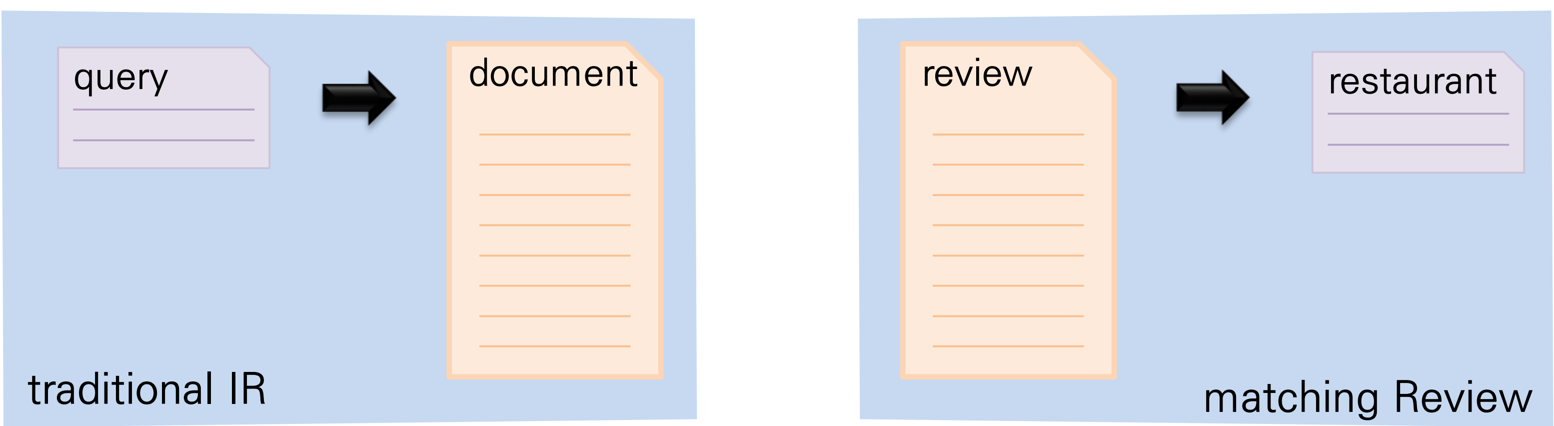
Two Settings of Related Flavor

- Entity Matching
  - to find the correspondence between two structured objects
- Information Retrieval(IR)
  - to match unstructured short text against unstructured text

Classical IR Methods Doesn't Fit

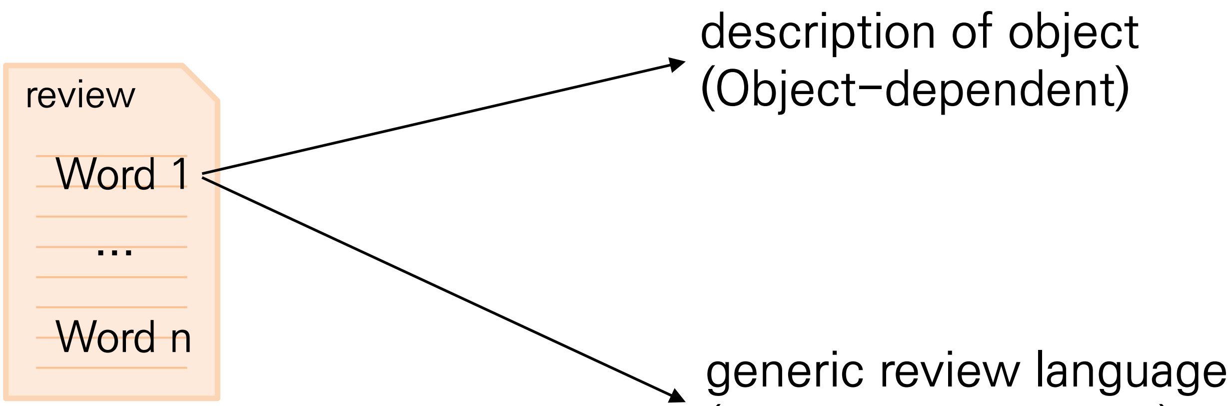
- Example of "Food"
  - "food" is rare as a restaurant name
  - thus, it will get a very high IDF score
  - AND hence will likely be the top match for all reviews containing the word "food"

- UNLIKE in traditional IR
  - a query (i.e. review) is long and a document (i.e. restaurant) is short



~~Our~~ Their Contributions

- 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(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

Review Language Model (RLM)

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

- It represent 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)

$$\begin{aligned}\Pr[r \mid e] &= Z(r) \prod_{w \in r} \Pr[w \mid e] \\ &= Z(r) \prod_{w \in r} ((1-\alpha)P(w) + \alpha P_e(w)), (1)\end{aligned}$$

It can be zero, if a word w is not in text(e)

Thus, have to modify the equation as following

$$\begin{aligned}\Pr[r \mid e] &= Z(r) \prod_{w \in r \setminus r_e} (1-\alpha)P(w) \cdot \prod_{w \in r_e} ((1-\alpha)P(w) + \alpha P_e(w)) \\ &= Z(r) \prod_{w \in r} (1-\alpha)P(w) \cdot \prod_{w \in r_e} \left(1 + \frac{\alpha}{1-\alpha} \frac{P_e(w)}{P(w)}\right). \qquad (2)\end{aligned}$$

Review Language Model (RLM)

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

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

$$e^* = \arg\max_e \sum_{w \in r_e} \log\left(1 + \frac{\alpha}{1-\alpha}\frac{P_e(w)}{P(w)}\right). \tag{3}$$

Review Language Model (RLM)

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

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

How?

$$e^* = \arg\max_e \sum_{w \in r_e} \log \left(1 + \frac{\alpha}{1-\alpha} \boxed{\frac{P_e(w)}{P(w)}}\right). \quad (3)$$

Review Language Model (RLM)

- Object-independent factor

$$P(w) = \frac{c(w, \mathcal{R}^{(g)}) + 1}{\sum_{w'} c(w', \mathcal{R}^{(g)}) + |V|},$$

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

- Object-dependent factor

(say, $g(w) = \log(1/f_w)$), we let

$$P_e(w) = \frac{g(w)}{\sum_{w' \in \mathrm{text}(e)} g(w')}.$$

  – By using the frequency f_w of the word w in $\mathcal{R}$ or in $\{\mathrm{text}(e) \mid e \in \mathcal{E}\}$.

Model and Method

RLM, TFIDF and TFIDF+

- Generic equation

$$e^* = \arg\max_{e} \sum_{w \in r_e} \log f(w)$$

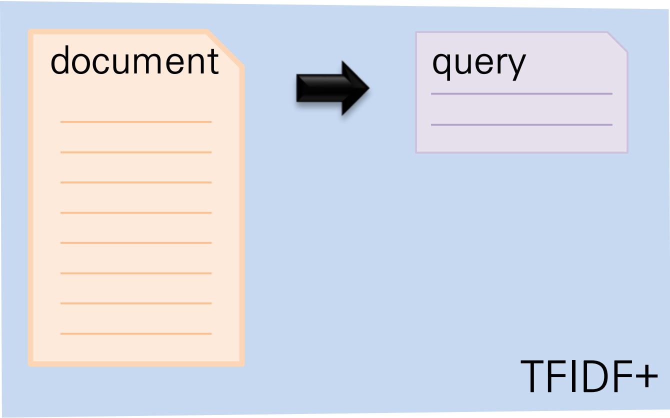
- for RLM, f(w) goes

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

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

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

$$Q(w) = \frac{\mathrm{df}(w)}{N}$$



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 restaurants
- 681,320 restaurants from Yahoo! Local database
- Task
  - to match a given Yelp review, using ONLY its free-form textual content

The Final Aligned Dataset

- $\mathcal{R}$
  - 24,910 Yelp reviews covering 6,010 restaurants
- $\mathcal{R}'$
  - to estimate the models
  - reviews filtered out because of lack of identifying information were added
  - 205,447 reviews
- $\mathcal{R}_{test}$
  - 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 $\mathcal{E}$
- 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

Main Result

Method	Micro-avg.	Macro-avg.
RLM	0.647	0.576
TFIDF$^+$	0.518	0.481
TFIDF	0.314	0.317

(a) Main comparison.

Method	Micro-avg.	Macro-avg.
RLM-UNIFORM	0.634	0.562
RLM-UNCUT	0.627	0.546
RLM-DECAP	0.640	0.573

(b) RLM variants.

Method	Micro-avg.	Macro-avg.
TFIDF$^+$-N	0.586	0.523
TFIDF$^+$-D	0.593	0.533
TFIDF$^+$-O	0.522	0.488
TFIDF$^+$-ND	0.628	0.549
TFIDF$^+$-NDO	0.647	0.576

(c) TFIDF$^+$ variants.

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

Main Result

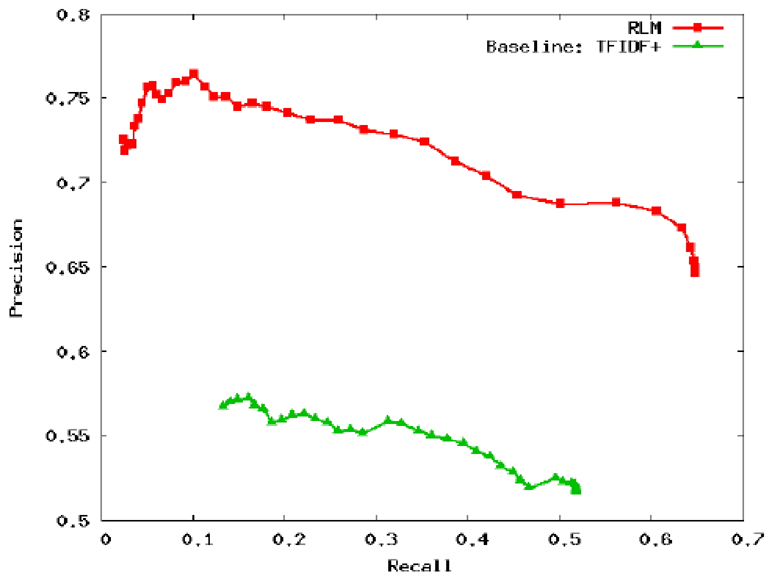


Figure 1: Precision–recall curve (of top one prediction): RLM vs. TFIDF$^+$ baseline.

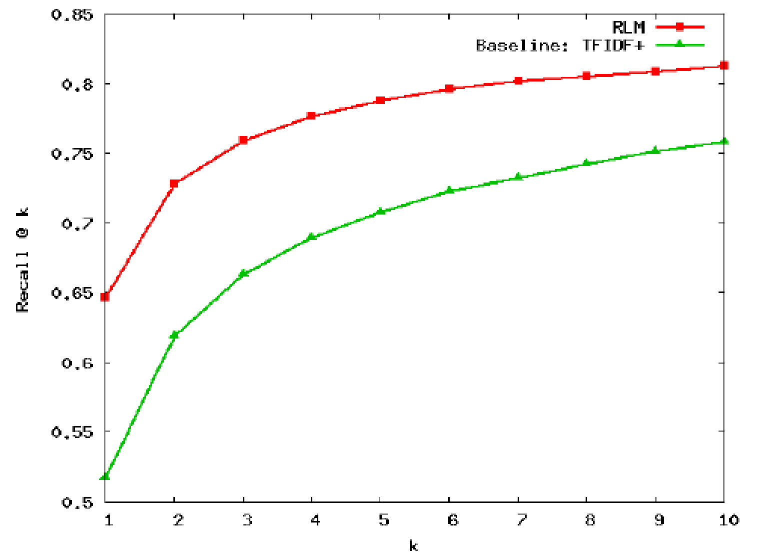


Figure 2: Accuracy@k (percentage of reviews whose correct match is returned in one of its top-k predictions): RLM vs. TFIDF$^+$ baseline.

Main Result

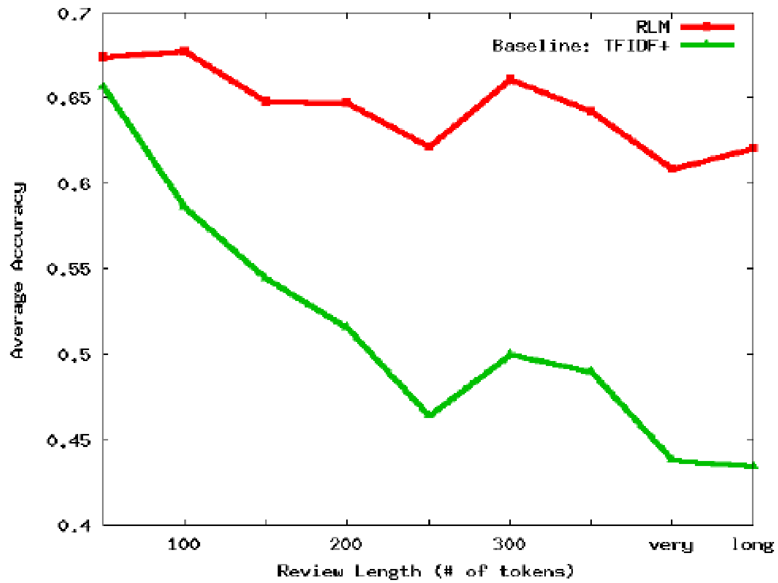


Figure 3: Average accuracy of the top-1 prediction for reviews with different length (on test set): RLM vs. TFIDF^{+} baseline.

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.

Main Result

Method	Micro-avg.	Macro-avg.
RLM	0.647	0.576
TFIDF+	0.518	0.481
TFIDF	0.314	0.317

(a) Main comparison.

Method	Micro-avg.	Macro-avg.
RLM-UNIFORM	0.634	0.562
RLM-UNCUT	0.627	0.546
RLM-DECAP	0.640	0.573

(b) RLM variants.

Method	Micro-avg.	Macro-avg.
TFIDF+-N	0.586	0.523
TFIDF+-D	0.593	0.533
TFIDF+-O	0.522	0.488
TFIDF+-ND	0.628	0.549
TFIDF+-NDO	0.647	0.576

(c) TFIDF+ variants.

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

- 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

Conclusions

- The model provides us a principled way to match reviews to objects
- Their techniques vastly outperforms standard TF-IDF based techniques