Translating Webpages into
Bidphrases for Advertising

Sujith Ravi*, Andrei Broder†, Evgeniy Gabrilovich†, Sandeep Pandey†, Bo Pang†, Vanja Josifovski†
sravi@isi.edu {broder, bopang, gabr, spandey, vanjaj}@yahoo-inc.com

* University of Southern California
Information Sciences Institute,
Los Angeles, CA

† Yahoo! Research,
Santa Clara, CA
The Birth of an Ad Campaign

Landing Page + Landing URL

http://www.scottrade.com/t?kwclid=TC-1391-tOVRWID-1powntc5.55Cl-1oVADID

Scottrade
Reliable. Stable. Trusted.
*Highest in Investor Satisfaction with Self-Directed Services*
See How We Compare

options first
Powerful tools for options trading.
Explore the Platform

FOR OUR CUSTOMERS
Money Direct - Same-day, electronic fund transfers from your bank account into your individual, joint, IRA, IRA, ESA or EDP account. Learn More
Scottrade Online Community - interact and learn from other Scottrade customers and product specialists. Join Today

Branch Locator
Find Your Branch
Enter Zip and Go

Contact Us by Phone or Email

Discover A Culture for Earning Success
The Birth of an Ad Campaign

Landing Page + Landing URL

Relevant Bidphrases

- scottrade best brokerage discount
- scottrade switch ira account
- scottrade stock information
- scottrade change ira fund
- scottrade best online brokerage firm
- scottrade best discount broker
- scottrade best online stock broker
- scottrade transfer ira fund
- scottrade switch ira fund
The Birth of an Ad Campaign

Can we automate this process?
Problem

webpage describing a product → Landing Page \((\ell)\) → Bid Phrases \((b)\) → relevant phrases on which advertiser can potentially bid
Problem

- Can you just get the most informative phrases in the page?
Problem

- Can you just get the most informative phrases in the page?

⇒ 96% of ads had at least one bid phrase not in $\ell$
Problem

- Can you just get the most informative phrases in the page?
  ➔ 96% of ads had at least one bid phrase not in \( \ell \)

- How about getting the words?
Can you just get the most informative phrases in the page?

- 96% of ads had at least one bid phrase not in $\ell$

How about getting the words?

- Need to mix-and-match in the right way to generate phrases
Problem

- Can you just get the most informative phrases in the page?
  - 96% of ads had at least one bid phrase not in \( l \)
- How about getting the words?
  - Need to mix-and-match in the right way to generate phrases
  - The bid phrase set for 70% of ads contained one or more words not in \( l \)
A Two-phase Approach

1. Candidate bid phrases are generated
   - need to be able to generate “novel” phrases

2. Candidates are ranked
   - need to pick phrases relevant to page and resemble queries
Translation-based Approach

Landing Page + Landing URL \( (l) \) \( \xrightarrow{\text{translation}} \) bidphrases \( (b) \)

- Noisy-channel approach used in Machine Translation

Generative Model

Bidphrase LM \( P(b) \) \( \xrightarrow{\text{bidphrases}} \) \( (b) \) \( \xrightarrow{b\text{-to-}l\ TM} \) landing page \( (l) \)
Translation-based Approach

Landing Page + Landing URL \( (l) \) \( \xrightarrow{\text{translation}} \) bidphrases \( (b) \)

- Noisy-channel approach used in Machine Translation

Generative Model

Bidphrase LM \( P(b) \) \( \xrightarrow{\text{bidphrases}} \) \( (b) \) \( \xrightarrow{\text{b-to-}l \text{ TM}} \) landing page \( (l) \)

Language Model generates potential bidphrases
Translation-based Approach

Landing Page + Landing URL \( (l) \) \( \xrightarrow{\text{translation}} \) bidphrases \( (b) \)

- Noisy-channel approach used in Machine Translation

Generative Model

- Bidphrase LM
  - Language Model generates potential bidphrases
  - \( P(b) \)

- Translation Model
  - Translates each bidphrase word \( (b_i) \) into a word \( (l_i) \) appearing on the landing page
  - \( P(l|b) \)

\( b \)-to-\( l \) Model

\( \text{Landing Page} + \text{Landing URL} \) \( \rightarrow \) bidphrases \( \rightarrow \) landing page

\( P(l|b) \)
Ranking Candidate Phrases

Given candidate bidphrases +

Bidphrase LM

\[ b \rightarrow \ell \]

TM

Score candidate bidphrases (Decoding)

landing page (\(\ell\))

candidate bidphrases \(\{b_1, b_2, b_3, ..., b_n\}\)

LM + TM

\[ P(b)P(l|b) \]

ranked list of bidphrases

\(b_1, \text{score}(b_1)\)

\(b_2, \text{score}(b_2)\)

\(b_n, \text{score}(b_n)\)
Bidphrase Language Model

Bidphrases should resemble queries

Estimating the model

- LM is a bigram language model, with back-off to a unigram model
- Model estimated on a large query corpus $Q$ (~76 million queries from Yahoo! Web search log)
Translation Model

Estimating the model

- Estimate translation table \( t(l_j \mid b_i) \) to maximize likelihood of (parallel) data (bid phrase, page) pairs

\[
Pr(l \mid b) \propto \prod_j \sum_i t(l_j \mid b_i)
\]

\( l_j = \) word in landing page \( l \)

\( b_i = \) word in bidphrase \( b \)
Translation Model

Estimating the model

- Estimate translation table \( t(l_j | b_i) \) to maximize likelihood of (parallel) data (bid phrase, page) pairs

\[
\Pr(l|b) \propto \prod_j \sum_i t(l_j | b_i)
\]

- Null token added to bidphrase side to account for irrelevant words from landing page

\[
l_j = \text{word in landing page } l
\]

\[
b_i = \text{word in bidphrase } b
\]
Translation Model

Estimating the model

- Estimate translation table $t(l_j \mid b_i)$ to maximize likelihood of (parallel) data (bid phrase, page) pairs

$$\Pr(l \mid b) \propto \prod_j \sum_i t(l_j \mid b_i)$$

- Null token added to bidphrase side to account for irrelevant words from landing page

- Null token added to bidphrase side to account for irrelevant words from landing page

- Incorporate importance of words in a page

$$\Pr(l \mid b) \propto \prod_j (\sum_i t(l_j \mid b_i))^{w_j}$$

$w_j = \text{importance weight assigned to word } l_j$

(higher weight for words appearing in titles, headings, etc.)
Generating Candidate Phrases

• Theoretically, all phrases in query log can be candidates ➔ inefficient

• **Strategy-1:** Build a candidate set containing only phrases appearing on landing page \( \{b_{LP}\} \)
  ➔ downside: no novel bidphrases generated

• **Strategy-2:** Use translation model (TM) to generate novel bidphrases \( \{b_{TMgen}\} \)
  ➔ bridges vocabulary mismatch
  ➔ use only *salient* words from landing page to generate new candidates
Alternative Methods

- Extraction-based system (Baseline)
  - extract candidates from page, rank by $\text{cosine-sim}(b, l)$
- Discriminative system using SVM\textsuperscript{rank} using features:
  - word-overlap, position on page, $\text{cosine-sim}(b, l)$, ....
- Using content-match system (CMS)

![Diagram]

landing page ($l$)

query vector

CMS

Ad corpus ($A$)

related bidphrases ($b$)

Best matched ads for $l$

ranked list of bidphrases

$\ b_1, \ \text{score}(b_1) \n
\ b_2, \ \text{score}(b_2) \n
\vdots \n
\ b_n, \ \text{score}(b_n) \n
Evaluation

Large-scale and automatic?

For each landing page \((\ell)\) in the test corpus (10,500 pages)

- Gold-standard bidphrases \(\{b_{\text{gold}}\}\)
  - provided by the advertisers
  - average 9 per landing page
- Each generated bidphrase \((b_c)\) is compared against \(\{b_{\text{gold}}\}\)

Relative ordering should be meaningful
Evaluation Metrics

1. Minimum Edit Distance (minED)

\[ \text{minED}(b_c, l) = \min_{b_j \in \{b_{gold}\}} \text{ED}(b_c, b_j) \]

where,

\[ \text{ED}(b_c, b_j) = \frac{\# \text{ of oprns. to convert } b_c \rightarrow b_j}{\# \text{ of words in } b_j} \]

lower minED scores => better

2. ROUGE-1 metric

\[ \text{ROUGE-1}(b_c, l) = \frac{\sum_{b_j \in \{b_{gold}\}} \# \text{ of words in } b_c \cap b_j}{\sum_{b_j \in \{b_{gold}\}} \# \text{ of words in } b_j} \]

higher ROUGE-1 scores => better

\( b_c \) against \( \{b_{gold}\} \) of page \( l \)
Evaluation Metrics

1. Minimum Edit Distance (minED)

\[ \text{minED}(b_c, l) = \min_{b_j \in \{b_{gold}\}} \text{ED}(b_c, b_j) \]

where,

\[ \text{ED}(b_c, b_j) = \frac{\text{\# of oprns. to convert } b_c \rightarrow b_j}{\text{\# of words in } b_j} \]

lower minED scores => better

2. ROUGE-1 metric

\[ \text{ROUGE-1}(b_c, l) = \frac{\sum_{b_j \in \{b_{gold}\}} \text{\# of words in } b_c \cap b_j}{\sum_{b_j \in \{b_{gold}\}} \text{\# of words in } b_j} \]

higher ROUGE-1 scores => better
Evaluation Metrics

\( b_c \) against \( \{b_{gold}\} \) of page \( l \)

1. **Minimum Edit Distance (minED)**

\[
minED(b_c, l) = \min_{b_j \in \{b_{gold}\}} ED(b_c, b_j)
\]

where,

\[
ED(b_c, b_j) = \frac{\# \text{ of oprns. to convert } b_c \rightarrow b_j}{\# \text{ of words in } b_j}
\]

lower \( minED \) scores => better

Is it similar to any phrase in \( \{b_{gold}\} \) ?

2. **ROUGE-1 metric**

\[
ROUGE-1(b_c, l) = \frac{\sum_{b_j \in \{b_{gold}\}} \# \text{ of words in } b_c \cap b_j}{\sum_{b_j \in \{b_{gold}\}} \# \text{ of words in } b_j}
\]

recall of words in \( \{b_{gold}\} \)

higher \( ROUGE-1 \) scores => better
Main Comparisons

\{b_{LP}\}

\{b_{LP+CMS}\}

\{b_{LP+TM_{gen}}\}

Candidate Generation
Main Comparisons

\[ \{ b_{LP} \} \rightarrow \text{words/phrases extracted from landing page} \]

\[ \{ b_{LP+CMS} \} \]

\[ \{ b_{LP+TM_{gen}} \} \]
Main Comparisons

\[ \{ b_{LP} \} \rightarrow \text{words/phrases extracted from landing page} \]

\[ \{ b_{LP+CMS} \} \rightarrow + \text{bidphrases proposed by CMS} \]

\[ \{ b_{LP+TM_{gen}} \} \]
Main Comparisons

\[ \{ b_{LP} \} \rightarrow \text{words/phrases extracted from landing page} \]

\[ \{ b_{LP+CMS} \} \]

\[ \{ b_{LP+TM_{gen}} \} \rightarrow + \text{new phrases generated by translating landing page content using TM} \]
Main Comparisons

Candidate Generation

\[
\begin{align*}
\{b_{LP}\} \\
\{b_{LP+CMS}\} \\
\{b_{LP+TM_{gen}}\}
\end{align*}
\]

Candidate Ranking
## Main Comparisons

<table>
<thead>
<tr>
<th>Candidate Generation</th>
<th>Candidate Ranking</th>
</tr>
</thead>
<tbody>
<tr>
<td>( {b_{LP}} )</td>
<td>cosine</td>
</tr>
<tr>
<td>( {b_{LP+CMS}} )</td>
<td>baseline</td>
</tr>
<tr>
<td>( {b_{LP+TM_{gen}}} )</td>
<td></td>
</tr>
</tbody>
</table>
# Main Comparisons

<table>
<thead>
<tr>
<th>Candidate Generation</th>
<th>Candidate Ranking</th>
</tr>
</thead>
<tbody>
<tr>
<td>( { b_{LP} } )</td>
<td>baseline</td>
</tr>
<tr>
<td>( { b_{LP+CMS} } )</td>
<td>CMS</td>
</tr>
<tr>
<td>( { b_{LP+TM_{gen}} } )</td>
<td>CMS</td>
</tr>
</tbody>
</table>

<table>
<thead>
<tr>
<th>cosine</th>
<th>CMS</th>
</tr>
</thead>
</table>
### Main Comparisons

<table>
<thead>
<tr>
<th>Candidate Generation</th>
<th>Candidate Ranking</th>
</tr>
</thead>
<tbody>
<tr>
<td>cosine</td>
<td>CMS</td>
</tr>
<tr>
<td>{b_{LP}}</td>
<td>baseline</td>
</tr>
<tr>
<td>{b_{LP+CMS}}</td>
<td>CMS</td>
</tr>
<tr>
<td>{b_{LP+TM_{gen}}}</td>
<td></td>
</tr>
</tbody>
</table>

- **Candidate Generation**
  - \{b_{LP}\}
  - \{b_{LP+CMS}\}
  - \{b_{LP+TM_{gen}}\}

- **Candidate Ranking**
  - cosine
  - CMS
  - SVM\textit{rank}
## Main Comparisons

<table>
<thead>
<tr>
<th>Candidate Generation</th>
<th>Candidate Ranking</th>
</tr>
</thead>
<tbody>
<tr>
<td>( { b_{LP} } )</td>
<td>baseline</td>
</tr>
<tr>
<td>( { b_{LP+CMS} } )</td>
<td>CMS</td>
</tr>
<tr>
<td>( { b_{LP+TM_{gen}} } )</td>
<td></td>
</tr>
<tr>
<td>cosine</td>
<td>CMS</td>
</tr>
</tbody>
</table>
Main Comparisons

**lower minED scores => better bidphrases**

<table>
<thead>
<tr>
<th>Baseline (cosine)</th>
<th>CMS</th>
<th>Discriminative System (SVM(^{rank}) with features)</th>
<th>LM+TM (B_{LP+TM_{gen}})</th>
</tr>
</thead>
<tbody>
<tr>
<td>minED @ rank 1</td>
<td>0.66</td>
<td>0.78</td>
<td>0.67</td>
</tr>
<tr>
<td>minED @ rank 5</td>
<td>0.71</td>
<td>0.81</td>
<td>0.72</td>
</tr>
<tr>
<td>minED @ rank 10</td>
<td>0.75</td>
<td>0.83</td>
<td>0.74</td>
</tr>
<tr>
<td>ROUGE-1 @ rank 1</td>
<td>0.24</td>
<td>0.22</td>
<td>0.26</td>
</tr>
<tr>
<td>ROUGE-1 @ rank 5</td>
<td>0.19</td>
<td>0.21</td>
<td>0.24</td>
</tr>
<tr>
<td>ROUGE-1 @ rank 10</td>
<td>0.16</td>
<td>0.20</td>
<td>0.22</td>
</tr>
</tbody>
</table>

**higher ROUGE-1 scores => better bidphrases**

Test corpus = 10,500 landing pages
Main Comparisons

<table>
<thead>
<tr>
<th></th>
<th>cosine</th>
<th>CMS</th>
<th>SVM\textsuperscript{rank}</th>
<th>LM+TM</th>
</tr>
</thead>
<tbody>
<tr>
<td>${b_{LP}}$</td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>${b_{LP+CMS}}$</td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>${b_{LP+TM_{gen}}}$</td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
</tbody>
</table>
## Candidate Generation

<table>
<thead>
<tr>
<th></th>
<th>cosine</th>
<th>CMS</th>
<th>SVM\textsuperscript{rank}</th>
<th>LM+TM</th>
</tr>
</thead>
<tbody>
<tr>
<td>({b_{LP}})</td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>({b_{LP}+CMS})</td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>({b_{LP}+TM_{gen}})</td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
</tbody>
</table>

Test corpus = 10,500 landing pages
## Ranking Methods

Test corpus = 10,500 landing pages

<table>
<thead>
<tr>
<th></th>
<th>cosine</th>
<th>CMS</th>
<th>SVM\text{\textsuperscript{rank}}</th>
<th>LM+TM</th>
</tr>
</thead>
<tbody>
<tr>
<td>{b_{LP}}</td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>{b_{LP+CMS}}</td>
<td>yellow</td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>{b_{LP+T_{Mgen}}}</td>
<td></td>
<td></td>
<td></td>
<td>red</td>
</tr>
</tbody>
</table>
Component Analysis for TM+LM

Varying the Language Model (unigram vs. bigram LM)

Varying the Translation Model (using different training sizes)

improvement from using better LM

improvement from using better TM
## Translation Table

| bidphrase word \((b_i)\) | top translations \(l_j\) | \(P(l_i|b_j)\) |
|-------------------------|--------------------------|-----------------|
| account                 | account                  | 0.756           |
|                         | accounts                 | 0.023           |
|                         | checking                 | 0.012           |
|                         | savings                  | 0.010           |
|                         | online                   | 0.009           |
| addiction               | addiction                | 0.940           |
|                         | drug                     | 0.012           |
|                         | alcohol                  | 0.009           |
|                         | rehab                    | 0.007           |
|                         | addict                   | 0.005           |
|                         | mag                      | 0.870           |
|                         | magazine                 | 0.025           |
|                         | cover                    | 0.013           |
|                         | subscription             | 0.009           |
|                         | magazin                  | 0.008           |
|                         | ticket                   | 0.288           |
|                         | tickets                  | 0.220           |
|                         | flights                  | 0.037           |
|                         | prices                   | 0.030           |
|                         | fares                    | 0.020           |
Related Work

• Online Advertising
  • keyword extraction [Yih et al., 2006]
  • bridging vocabulary overlap in contextual advertising [Ribeiro-Neto et al., 2005]
  • query expansion and rewriting, keyword suggestion, ...
• Machine Translation / noisy channel model
  • text summarization [Knight and Marcu, 2000]; paraphrase extraction [Quirk et al., 2004]
  • contextual advertising [Murdock et al., 2007]
Conclusion

• Several automatic methods to generate bidphrases for online advertising

• Two evaluation measures proposed to assess different qualitative aspects of generated bidphrases

• Novel translation-based approach using a generative model
  ➡ produces best results in terms of both evaluation measures
  ➡ generates novel phrases that are relevant but do not appear on page