Translating Webpages into Bidphrases for Advertising

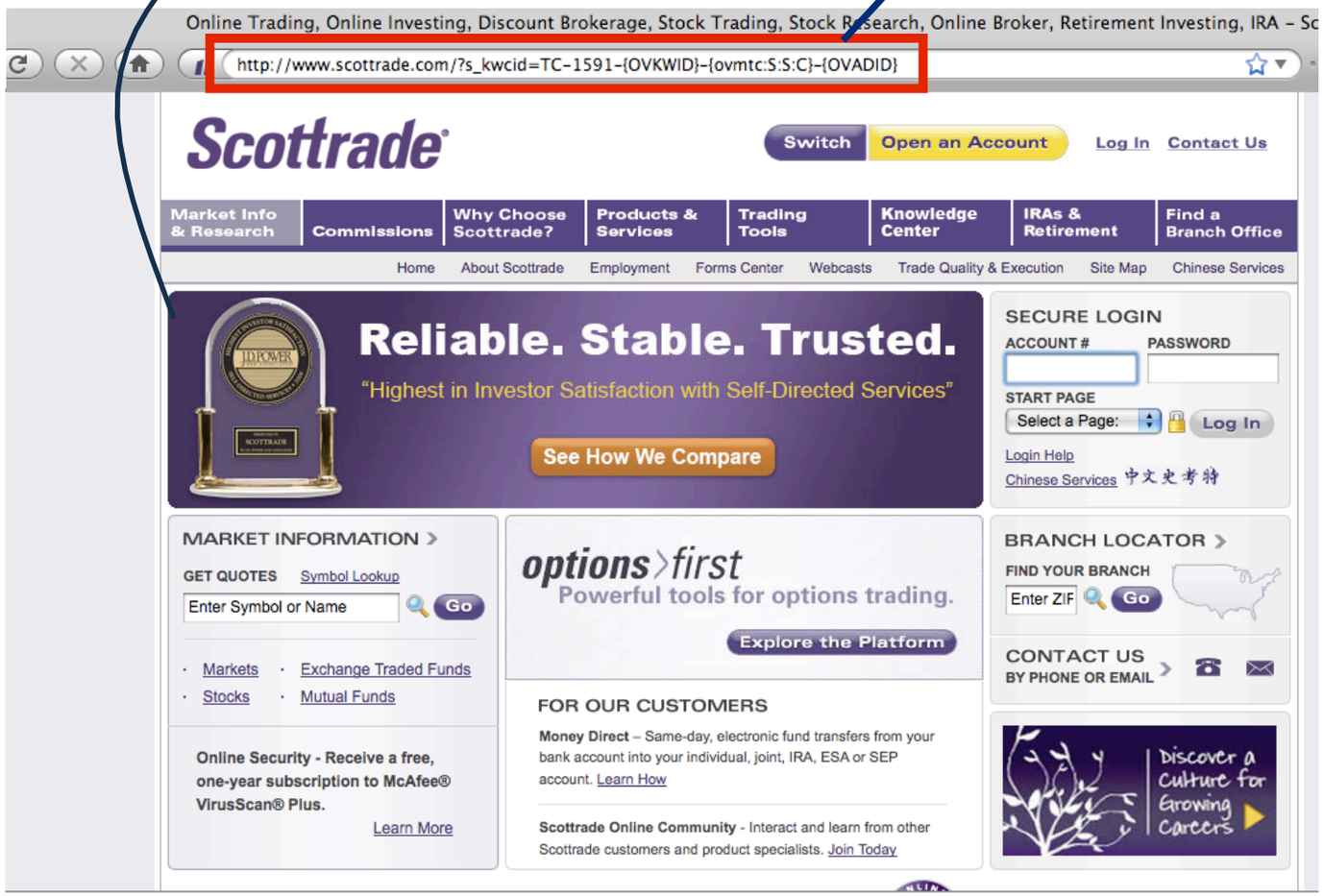
Sujith Ravi*, Andrei Broder†, Evgeniy Gabrilovich†,
Sandeep Pandey†, *Bo Pang*†, Vanja Josifovski†
sravi@isi.edu {*broder, bopang, gabr, spandey, vanjaj*}*@yahoo-inc.com*

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

† **Yahoo! Research**,
Santa Clara, CA

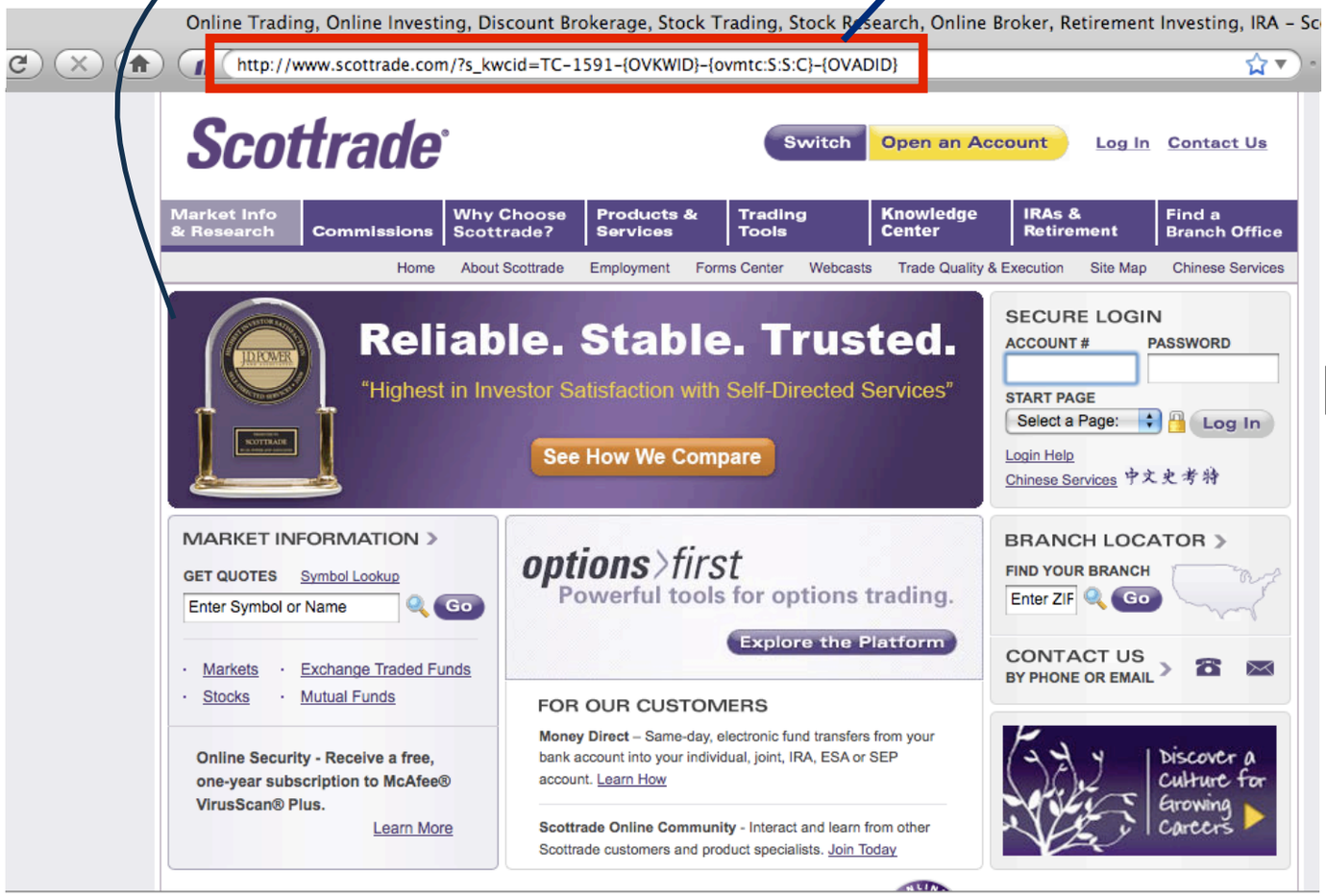
The Birth of an Ad Campaign

Landing Page + Landing URL



The Birth of an Ad Campaign

Landing Page + Landing URL

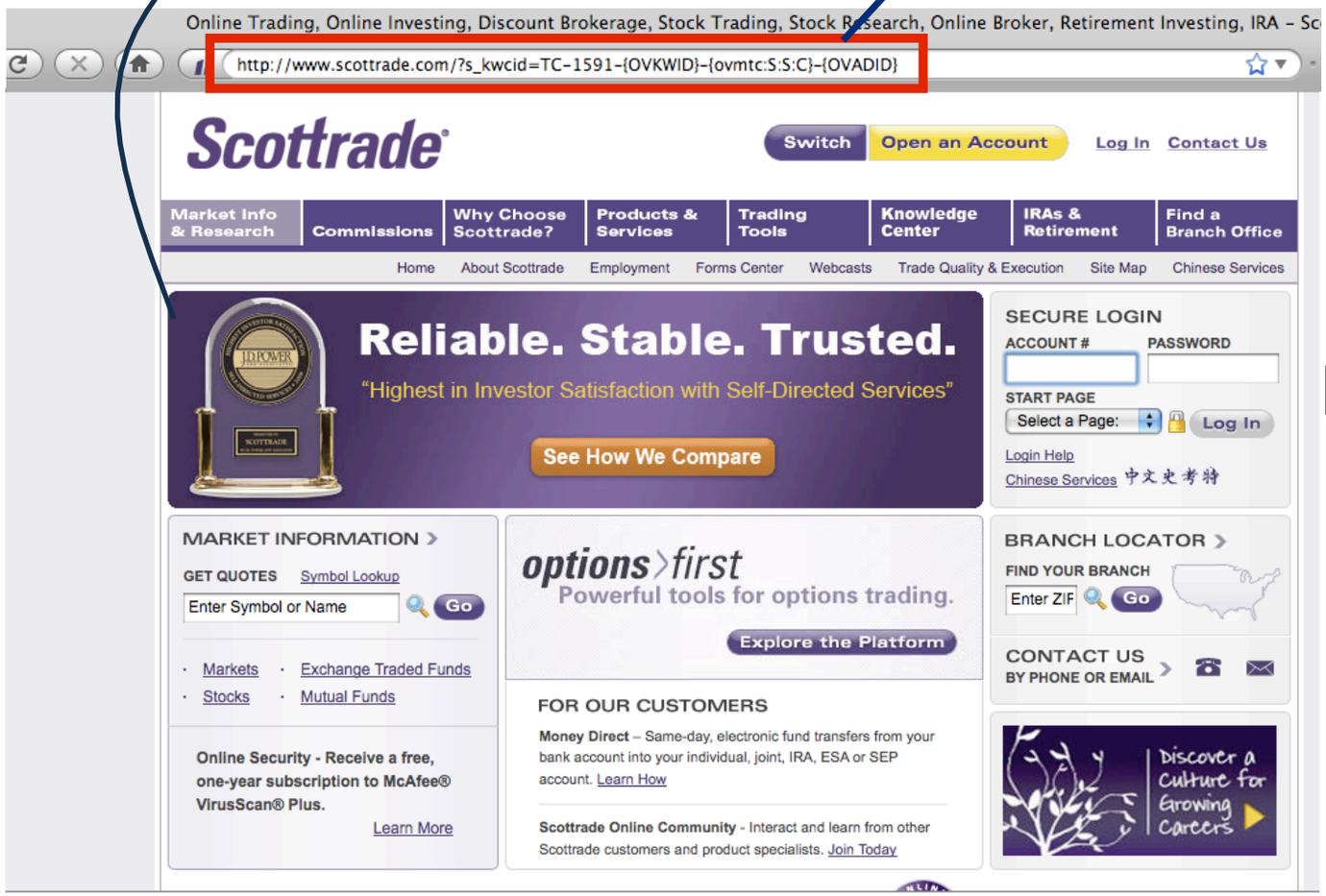


Relevant Bidphrases

```
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

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
```

Can we automate this process?

Problem

webpage describing a product

Landing Page
(ℓ)

Bid Phrases
(b)

relevant phrases on which advertiser can potentially bid

Problem

webpage describing a product

Landing Page
(ℓ)

Bid Phrases
(b)

relevant phrases on which advertiser can potentially bid

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

Problem

webpage describing a product

Landing Page (ℓ) ⟶ Bid Phrases (b)

relevant phrases on which advertiser can potentially bid

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

➡ 96% of ads had at least one bid phrase not in ℓ

Problem

webpage describing a product

Landing Page (ℓ) ⟶ Bid Phrases (b)

relevant phrases on which advertiser can potentially bid

- 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?

Problem

webpage describing a product

Landing Page (ℓ) ⟶ Bid Phrases (b)

relevant phrases on which advertiser can potentially bid

- 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*

Problem

webpage describing a product

Landing Page (ℓ) ⟶ Bid Phrases (b)

relevant phrases on which advertiser can potentially bid

- 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

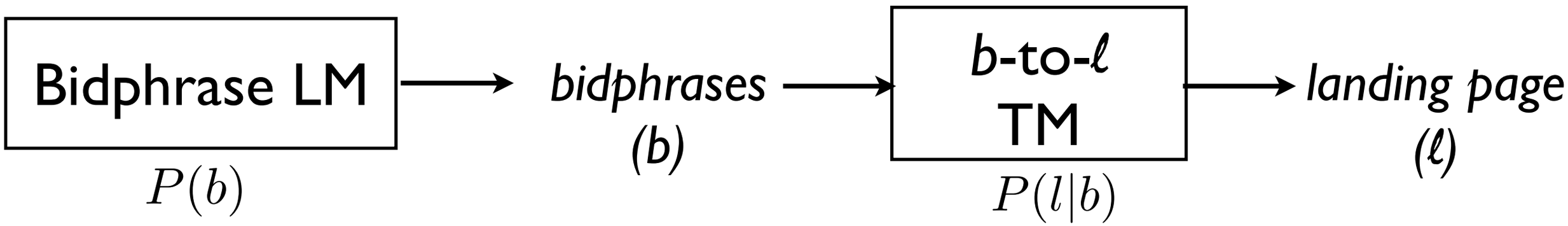
1. Candidate bid phrases are generated
    - need to be able to generate "novel" phrases
2. Candidates are ranked
    - need to pick phrases relevant to page and resemble queries

Translation-based Approach

Landing Page + Landing URL *(l)* —**translation**→ bidphrases *(b)*

- Noisy-channel approach used in Machine Translation

Generative Model

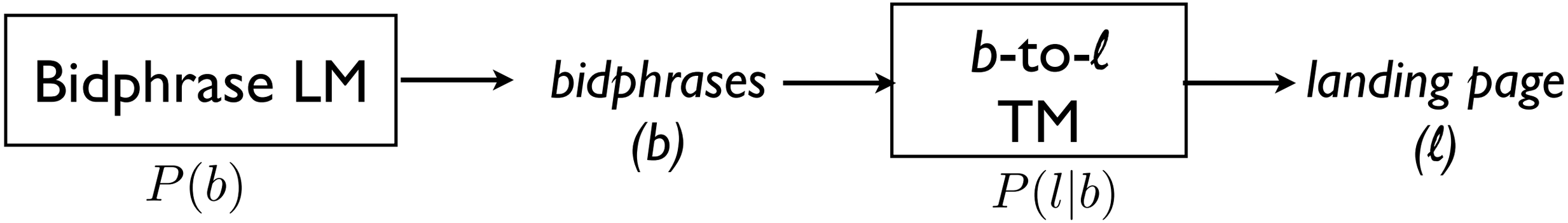


Translation-based Approach

Landing Page + Landing URL *(l)* —**translation**→ bidphrases *(b)*

- Noisy-channel approach used in Machine Translation

Generative Model



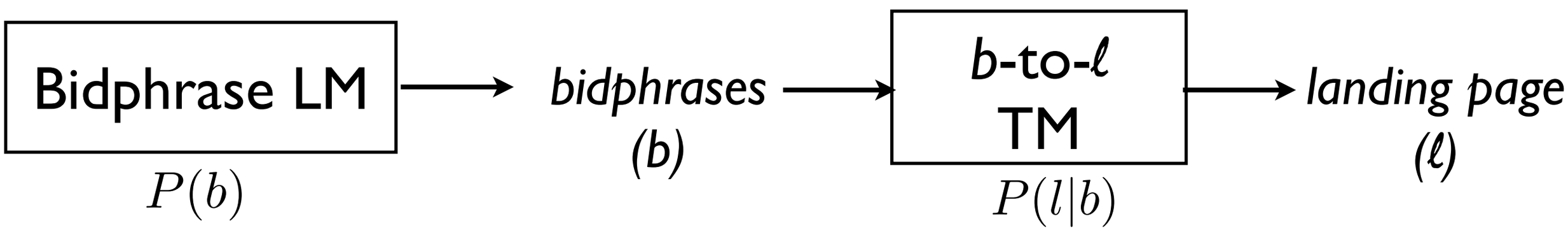
Language Model generates potential bidphrases

Translation-based Approach

Landing Page + Landing URL *(l)* —**translation**→ bidphrases *(b)*

- Noisy-channel approach used in Machine Translation

Generative Model



Language Model generates potential bidphrases

Translation Model translates each bidphrase word (b_i) into a word (ℓ_i) appearing on the landing page

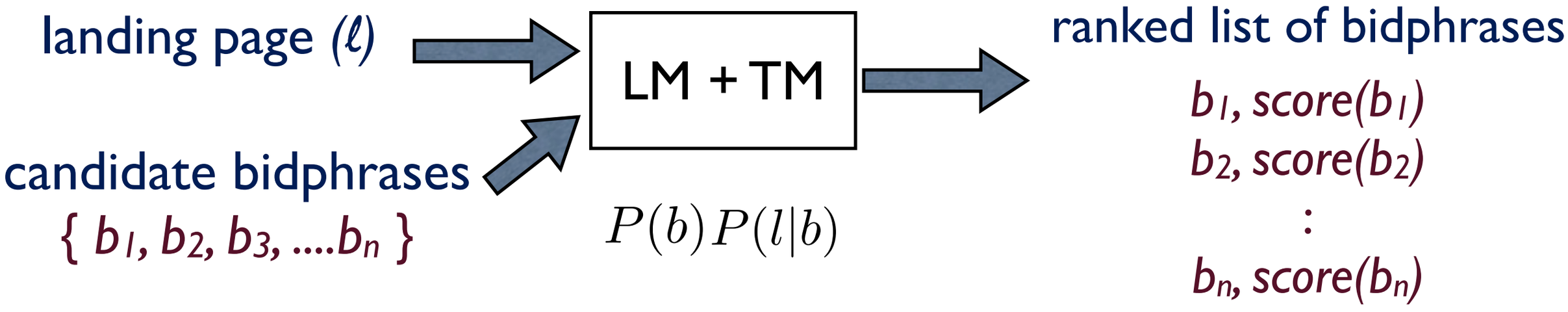
Ranking Candidate Phrases

Given candidate bidphrases +

Bidphrase LM

b-to-*ℓ* TM

Score candidate bidphrases (Decoding)



Bidphrase Language Model LM

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 TM

Estimating the model

IBM Model 1

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

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

l_j = word in landing page l
b_i = word in bidphrase b

Translation Model TM

Estimating the model

IBM Model 1

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

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

l_j = word in landing page l
b_i = word in bidphrase b

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

Translation Model TM

Estimating the model

IBM Model 1

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

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

l_j = word in landing page l
b_i = word in bidphrase b

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

- Incorporate importance of words in a page

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

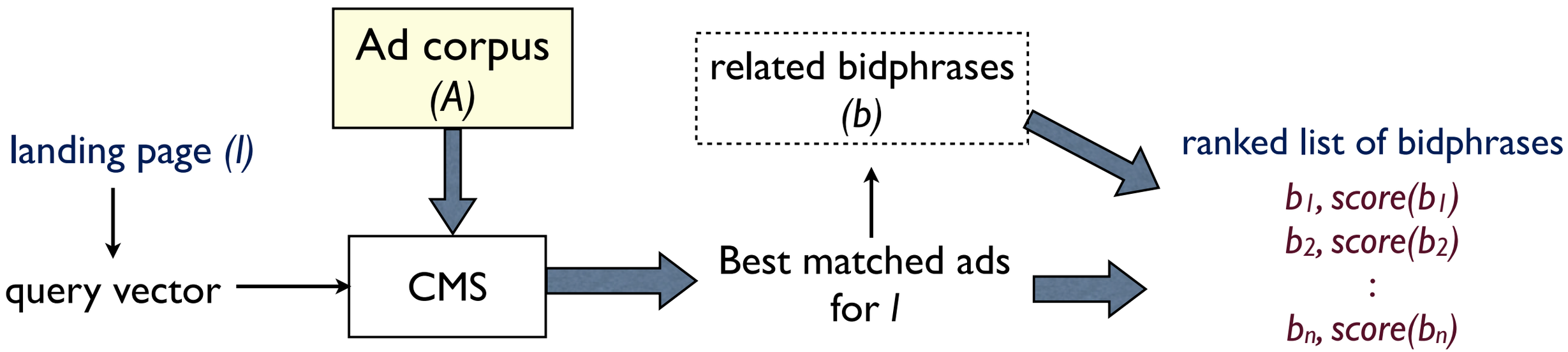
w_j = importance weight assigned to word l_j
(higher weight for words appearing in titles, headings, etc.)

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 *cosine-sim(b, ℓ)*
- Discriminative system using SVM^{rank} using features:
  - *word-overlap, position on page, cosine-sim(b, ℓ), ...*
- Using content-match system (CMS)



Evaluation

Large-scale and automatic?

For each landing page *(ℓ)* in the test corpus (10,500 pages)

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

Relative ordering should be meaningful

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

2. ROUGE-1 metric

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

higher *ROUGE-1* scores => better

Evaluation Metrics

b_c against $\{b_{gold}\}$ of page ℓ

1. Minimum Edit Distance *(minED)*

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

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

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

2. ROUGE-1 metric

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

higher *ROUGE-1* scores => better

Evaluation Metrics

b_c against $\{b_{gold}\}$ of page ℓ

1. Minimum Edit Distance *(minED)*

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

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

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

2. ROUGE-1 metric

recall of words in $\{b_{gold}\}$

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

higher *ROUGE-1* scores => better

Main Comparisons

$\{b_{LP}\}$

$\{b_{LP+CMS}\}$

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

Candidate Generation

Main Comparisons

$\{b_{LP}\}$ ➡ words/phrases extracted from landing page

$\{b_{LP+CMS}\}$

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

Candidate Generation

Main Comparisons

$\{b_{LP}\}$ ➡ words/phrases extracted from landing page

$\{b_{LP+CMS}\}$ ➡ + bidphrases proposed by CMS

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

Candidate Generation

Main Comparisons

Candidate Generation

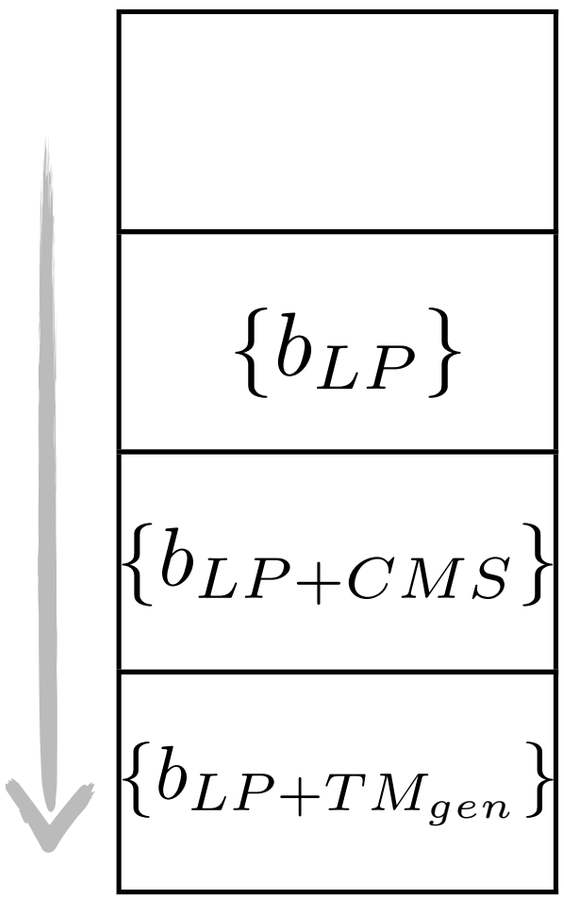
$\{b_{LP}\}$ ➡ words/phrases extracted from landing page

$\{b_{LP+CMS}\}$

$\{b_{LP+TM_{gen}}\}$ ➡ + new phrases generated by translating landing page content using TM

Main Comparisons

Candidate Ranking



Candidate Generation

Main Comparisons

Candidate Ranking →

	cosine
$\{b_{LP}\}$	baseline
$\{b_{LP+CMS}\}$	
$\{b_{LP+TM_{gen}}\}$	

↓ Candidate Generation

Main Comparisons

Candidate Ranking →

	cosine	CMS
$\{b_{LP}\}$	baseline	
$\{b_{LP+CMS}\}$		CMS
$\{b_{LP+TM_{gen}}\}$		

↓ Candidate Generation

Main Comparisons

Candidate Ranking →

	cosine	CMS	SVM^{rank}
$\{b_{LP}\}$	baseline		
$\{b_{LP+CMS}\}$		CMS	Discriminative system
$\{b_{LP+TM_{gen}}\}$			

↓ Candidate Generation

Main Comparisons

Candidate Ranking →

	cosine	CMS	SVM^{rank}	LM+TM
$\{b_{LP}\}$	baseline			
$\{b_{LP+CMS}\}$		CMS	Discriminative system	
$\{b_{LP+TM_{gen}}\}$				Translation -based

↓ Candidate Generation

Main Comparisons

lower *minED* scores => better bidphrases

	Baseline (cosine)	CMS	Discriminative System (SVM^{rank} with features)	LM+TM $B_{LP+TM_{gen}}$
minED @ rank 1	0.66	0.78	0.67	0.68
minED @ rank 5	0.71	0.81	0.72	0.68
minED @ rank 10	0.75	0.83	0.74	0.70
ROUGE-1 @ rank 1	0.24	0.22	0.26	0.29
ROUGE-1 @ rank 5	0.19	0.21	0.24	0.28
ROUGE-1 @ rank 10	0.16	0.20	0.22	0.27

higher *ROUGE-1* scores => better bidphrases

Test corpus = 10,500 landing pages

Main Comparisons

	cosine	CMS	SVM*rank*	LM+TM
$\{b_{LP}\}$				
$\{b_{LP+CMS}\}$				
$\{b_{LP+TM_{gen}}\}$				

Test corpus = 10,500 landing pages

Candidate Generation

	cosine	CMS	SVM*rank*	LM+TM
$\{b_{LP}\}$				
$\{b_{LP+CMS}\}$				
$\{b_{LP+TM_{gen}}\}$				

Test corpus = 10,500 landing pages

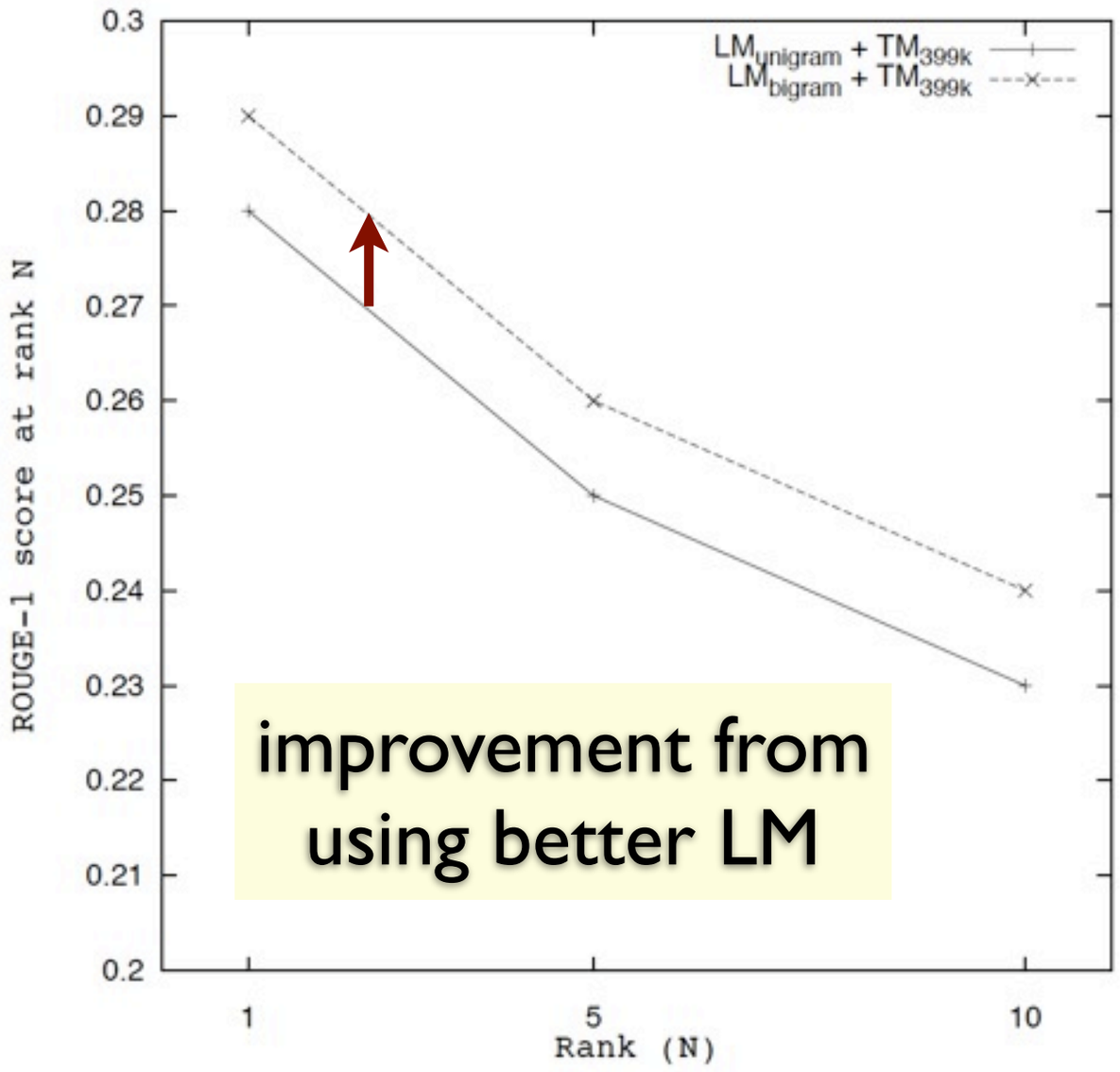
Ranking Methods

	cosine	CMS	SVM*rank*	LM+TM
$\{b_{LP}\}$				
$\{b_{LP+CMS}\}$				
$\{b_{LP+TM_{gen}}\}$				

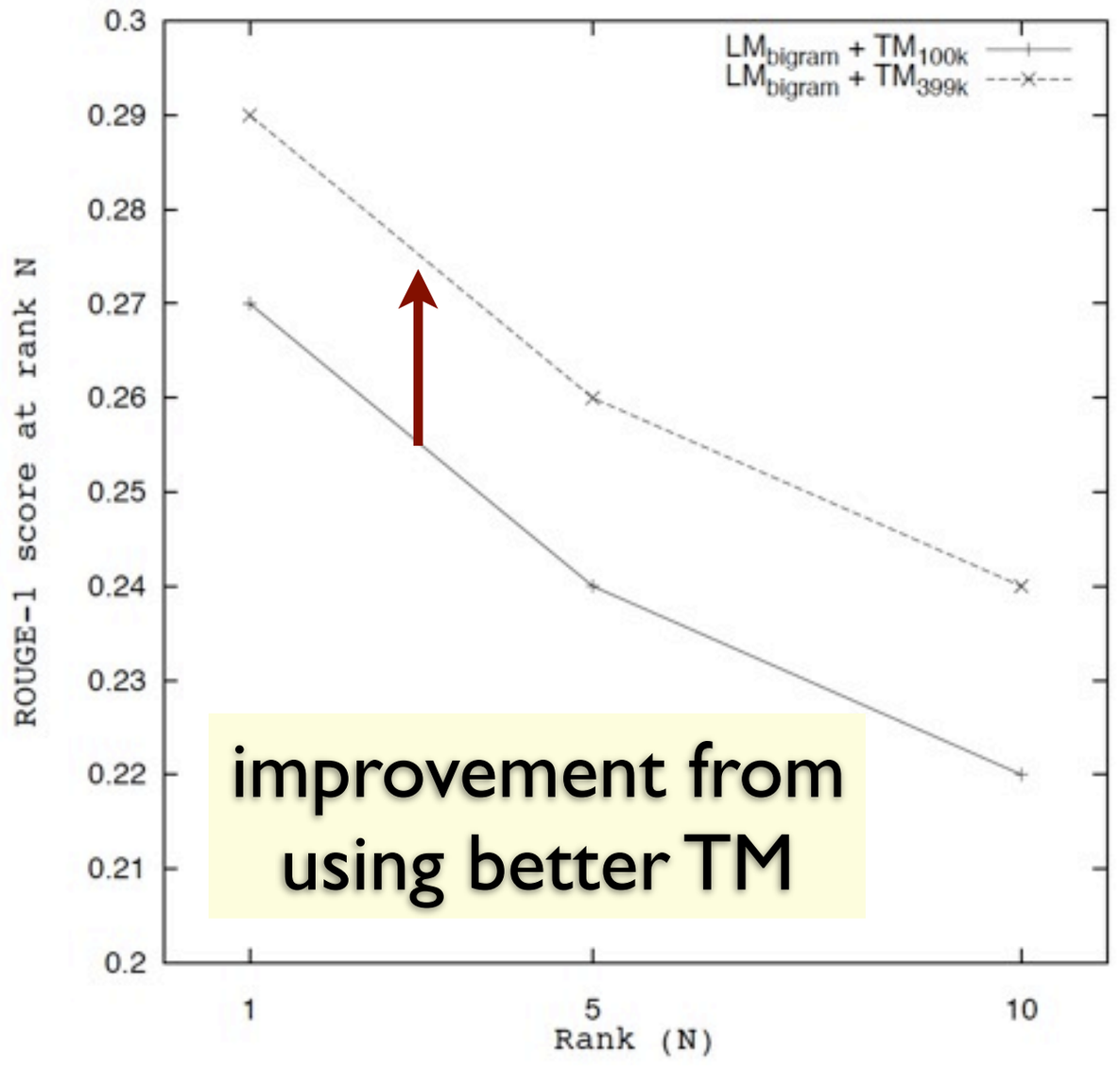
Test corpus = 10,500 landing pages

Component Analysis for TM+LM

Varying the Language Model (unigram vs. bigram LM)



Varying the Translation Model (using different training sizes)



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

bidphrase word (b_i)	top translations l_j	$P(l_i\|b_j)$
mag	*mag*	0.870
	magazine	0.025
	cover	0.013
	subscription	0.009
	magazin	0.008
ticket	*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]
  - contexual 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