PARAPHRASING ADAPTATION FOR WEB SEARCH RANKING

Chenguang Wang (Peking University)
Nan Duan (Microsoft Research Asia)
Ming Zhou (Microsoft Research Asia)
Ming Zhang (Peking University)

August 5, 2013
Mismatch between queries and documents is a key issue for the web search task

- Caused by expressing the same meaning in different natural language ways
  - E.g.
    - X is the author of Y
    - Y was written by X

Who is the author of *Gone with the Wind*?

Paraphrases

*Gone with the Wind* was written by whom?
MOTIVATION

Mismatch between queries and documents is a key issue for the web search task

• Caused by expressing the same meaning in different natural language ways
  • E.g.
    X is the author of Y
    Y was written by X

Paraphrasing engine produces alternative expressions to convey the same meaning of the input text

• Improve paraphrasing from different perspectives
  • E.g.
    Paraphrase extraction
    Paraphrase generation
    Model optimization
MOTIVATION (CONT.)

Q1: Could paraphrasing engine alleviate the mismatches of query and its relevant documents?

Q2: How to adapt the paraphrasing engine for web search ranking task specifically?
Solution Overview
Solution Overview

Paraphrase Extraction
- Extract paraphrase pairs from various data sources

Raw Data

Paraphrase Extraction
Solution Overview

Paraphrase Extraction
- Extract paraphrase pairs from various data sources

Paraphrase Model
- A search-oriented model generates candidates for each original query
Solution Overview

Paraphrase Extraction
- Extract paraphrase pairs from various data sources

Paraphrase Model
- A search-oriented model generates candidates for each original query

Parameter Optimization
- Optimize the weights of the features used in paraphrasing model on development data

\[ \sum_{i} \lambda_i \cdot h_i(\cdot) \]
Solution Overview

Paraphrase Extraction
- Extract paraphrase pairs from various data sources

Paraphrase Model
- A search-oriented model generates candidates for each original query

Parameter Optimization
- Optimize the weights of the features used in paraphrasing model on development data

Ranking Model
- An enhanced ranking model by using augmented features computed on paraphrases of original queries

Raw Data

Paraphrase Extraction

Paraphrase Model

Original Query + N-best Candidates

Original Query

Model Optimization

DEV Data

\[ \sum_i \lambda_i \cdot h_i(\cdot) \]
PARAPHRASE EXTRACTION

Monolingual-based

- Hypothesis:
  Words/Phrases that share the same context tend to have similar meanings
  \((\text{Lin and Pantel (2001)})\)

Bilingual-based

- Hypothesis:
  Phrases that align with identical pivot phrases tend to have similar meanings
  \((\text{Bannard and Callison-Burch (2005)})\)

#1 is the author of #2
#1 is #2 's author
SEARCH-ORIENTED PARAPHRASING MODEL

\[ \hat{Q} = \arg \max_{Q' \in \mathcal{H}(Q)} P(Q'|Q) \]

\[ = \arg \max_{Q' \in \mathcal{H}(Q)} \sum_{m=1}^{M} \lambda_m h_m(Q, Q') \]
SEARCH-ORIENTED PARAPHRASING MODEL

**Search-Oriented Features:**

- Word Addition
- Word Deletion
- Word Overlap
- Word Alteration
- Word Reordering
- Length Difference
- Edit Distance

\[
\hat{Q} = \arg \max_{Q' \in \mathcal{H}(Q)} P(Q'|Q)
\]

\[
= \arg \max_{Q' \in \mathcal{H}(Q)} \sum_{m=1}^{M} \lambda_m h_m(Q, Q')
\]

- **Candidate**
- **Original query**
- **Hypothesis space**

found a company

start a business
SEARCH-ORIENTED PARAPHRASING MODEL

Search-Oriented Features:

- Word Addition
- Word Deletion
- Word Overlap
- Word Alteration
- Word Reordering
- Length Difference
- Edit Distance

Traditional Features (Koehn et al., 2003):

- Translation Probability
- Lexical Weight
- Word Count
- Paraphrase Rule Count
- Language Model

\[ \hat{Q} = \arg \max_{Q' \in \mathcal{H}(Q)} P(Q' | Q) \]

\[ \hat{Q} = \arg \max_{Q' \in \mathcal{H}(Q)} \sum_{m=1}^{M} \lambda_m h_m(Q, Q') \]
NDCG-BASED PARAMETER OPTIMIZATION
NDCG-BASED PARAMETER OPTIMIZATION

Original Query
NDCG-BASED PARAMETER OPTIMIZATION

Original Query

Candidate-1
Candidate-2
...
Candidate-N
NDCG-BASED PARAMETER OPTIMIZATION

Original Query

Feature vector-1 → Candidate-1
Feature vector-2 → Candidate-2
...
Feature vector-N → Candidate-N
NDCG-BASED PARAMETER OPTIMIZATION

Feature vector-1 → Candidate-1 → Ranker
Feature vector-2 → Candidate-2 → Ranker
... → ... → ...
Feature vector-N → Candidate-N → Ranker

Original Query
NDCG-BASED PARAMETER OPTIMIZATION

Candidate is sent to the ranker, and returned by an NDCG score.
NDCG-BASED PARAMETER OPTIMIZATION

Original Query

Candidate-1 NDCG-1
Candidate-2 NDCG-2
... ...
Candidate-N NDCG-N

Candidate is sent to the ranker, and returned by an NDCG score

Ranker
... Ranker
... Ranker

Feature vector-1
Feature vector-2
... Feature vector-N

NDCG-based MER Training
NDCG-BASED PARAMETER OPTIMIZATION

Original Query

| Feature vector-1 | Candidate-1 | NDCG-1 |
|------------------|-------------|--------|
| Feature vector-2 | Candidate-2 | NDCG-2 |
| ...              | ...         | ...    |
| Feature vector-N | Candidate-N | NDCG-N |

Candidate is sent to the ranker, and returned by an NDCG score

Updated feature weights

\[ \sum_{i} \lambda_i \cdot h_i(\cdot) \]

After optimization, candidates with higher NDCGs are preferred and ranked on the top of the N-best list
NDCG-BASED PARAMETER OPTIMIZATION (CONT.)

Minimum error rate training (MERT) (Och, 2003)

- To find the optimal feature weight vector that minimizes the error criterion $Err$ according to the NDCG scores of top-1 paraphrase candidates

$$\hat{\lambda}_1^M = \arg\min_{\lambda_1^M} \left\{ \sum_{i=1}^{S} Err(D_i^{Label}, \hat{Q}_i; \lambda_1^M, \mathcal{R}) \right\}$$

$$Err(D_i^{Label}, \hat{Q}_i; \lambda_1^M, \mathcal{R}) = 1 - N(D_i^{Label}, \hat{Q}_i, \mathcal{R})$$
ENHANCED RANKING MODEL

Ranking model

• The paraphrase candidates act as hidden variables and expanded matching features between queries and documents

\[ R(Q, D_Q) = \sum_{k=1}^{K} \lambda_k F_k(Q, D_Q) \]

\[ \mathcal{R}(Q, D^i_Q) > \mathcal{R}(Q, D^j_Q) \iff r^i_{D_Q} > r^j_{D_Q} \]

- Unigram/bigram/trigram BM25
- Original/normalized Perfect-Match

Original query

\[ Q, Q'_1, Q'_2, \ldots, Q'_N \]

N-best paraphrase candidates

Retrieved documents

Document \( D_Q \)

\[ \overrightarrow{F'} = (F_1, F_2, \ldots, F_K) \]

\[ \{ \overrightarrow{F}, \overrightarrow{F_1}, \overrightarrow{F_2}, \ldots, \overrightarrow{F_N} \} \]
EXPERIMENTS: DATASETS

Paraphrase Extraction

- Training data
  - Bilingual corpus (NIST 2008 constrained track): 5.1M sentence pairs
  - Monolingual corpus (Bing’s query log): 16.7M queries
  - Human annotated data (WordNet dictionary): 0.3M synonym pairs
- # of paraphrase pairs: 58M

Evaluation Set

| Bing’s query log | # of queries |
|------------------|--------------|
| Development      | 1,419        |
| Test             | 1,419        |
**Paraphrasing**

| Denotation         | Features                                      | Optimization Metric |
|--------------------|-----------------------------------------------|---------------------|
| BL-Para (baseline) | Traditional features                          | BLEU                |
| BL-Para+SF         | Traditional features + Search-oriented features| BLEU                |
| BL-Para+SF+Opt     | Traditional features + Search-oriented features| NDCG                |

**Ranking Model**

| Denotation                  | Features                                                                 |
|-----------------------------|---------------------------------------------------------------------------|
| BL-Rank (baseline: Liu et al., 2007) | Query-documents matching features (unigram/bigram/trigram BM25 and original/normalized Perfect-Match) |
| BL-Rank+Para (Enhanced ranking model) | Query+Paraphrase-documents matching features |

*The ranking model is learned based on SVMrank toolkit (Joachims, 2006) with default parameter setting.*
# IMPACTS OF SEARCH-ORIENTED FEATURES

| Test Set   | BL-Para | BL-Para+SF |
|------------|---------|-----------|
| Original Query | Cand@1  | Cand@1    |
| 27.28%     | 26.44%  | 26.53%    |

**BL-Para:**
- Paraphrase Baseline with **Features:** Traditional features
- Optimization Metric: BLEU

**BL-Para+SF:**
- Paraphrase Baseline with **Features:** Traditional features + *Search-oriented features*
- Optimization Metric: BLEU
### IMPACTS OF OPTIMIZATION ALGORITHM

| Test Set        | BL-Para+SF | BL-Para+SF+Opt |
|-----------------|------------|---------------|
| Original Query  | Cand@1     | Cand@1        |
| 27.28%          | 26.53%     | 27.06% (+0.53%) |

**Top-1 Paraphrase Candidate**

**BL-Para+SF:**
Paraphrase Baseline with **Features:** Traditional features + Search-oriented features
Optimization Metric: **BLEU**

**BL-Para+SF+Opt:**
Paraphrase Baseline with **Features:** Traditional features + Search-oriented features
Optimization Metric: **NDCG**
## IMPACTS OF ENHANCED RANKING MODEL

### Ranking model baseline (Liu et al., 2007)

|               | Dev Set | Test set |
|---------------|---------|----------|
|               | NDCG@1  | NDCG@5   | NDCG@1  | NDCG@5   |
| BL-Rank       | 25.31%  | 33.76%   | 27.28%  | 34.79%   |
| BL-Rank+Para  | 28.59% (+3.28%) | 34.25% (+0.49%) | 28.42% (+1.14%) | 35.68% (+0.89%) |

### Enhanced ranking model

**BL-Rank:**

*Query-documents* matching features
(unigram/bigram/trigram BM25 and original/normalized Perfect-Match)

**BL-Rank+Para:**

*Query+Top 1 Paraphrase-documents* matching features
(unigram/bigram/trigram BM25 and original/normalized Perfect-Match)
CONCLUSION

We present an in-depth study on adapting paraphrasing for web search

• Paraphrasing model with search-oriented features
• NDCG-based optimization method

Future directions:

• Compare and combine paraphrasing with other query reformulation techniques to further improve the search quality
  • E.g., pseudo-relevance feedback, and conditional random field-based approach
THANK YOU!

QUESTIONS, EMAIL CHEN GUANG WANG
WANGCHEN@PKU.EDU.CN