Boosting Cross-Language Retrieval by Learning Bilingual Phrase Associations from Relevance Rankings

Artem Sokolov, Laura Jehl, Felix Hieber, Stefan Riezler
Cross-Lingual Information Retrieval: State-of-the-art

**Direct translation**
- translate query with SMT system
- monolingual retrieval with 1-best translation
- easy to deploy
- useful provided lots of in-domain data
Probabilistic structured queries

- query representation that includes translation alternatives
- estimate expected tf/idf weights & retrieve monolingually
- uses “good” n-best translations
- implicit query expansion by considering translation alternatives
**Drawbacks of standard approaches**

- crucial dependence on SMT quality
- SMT tuned for translation quality
- no learning for retrieval

**This paper**

- learns $n$-gram “phrase-table” relevant for the task
- optimizes final retrieval objective
- independent of any SMT system
- standalone: as good as a large domain-tuned SMT system or better
- combined with SMT baselines: +7 MAP & +15 PRES points.
Drawbacks of standard approaches

- crucial dependence on SMT quality
- SMT tuned for translation quality
- no learning for retrieval

This paper

- learns \( n \)-gram “phrase-table” relevant for the task
- optimizes final retrieval objective
- independent of any SMT system
- standalone: as good as a large domain-tuned SMT system or better
- combined with SMT baselines: +7 MAP & +15 PRES points.
Section 1

Baseline Approaches
Baseline Approaches

**Direct translation**

1. **SMT model for query translation**
   - state-of-the-art SCFG decoder (cdec) [Dyer 10]
   - word alignments from parallel data (mgiza++)
   - in-domain language model (kenlm) [Heafield 11]
   - parameter tuning with MERT [Och 03]

2. **Retrieval**
   - Okapi BM25 ranking
Probabilistic structured queries

(1) Query projection

- calculate expected \textit{tf/idf weights} with word translation probabilities:
  \[
  tf(f, E) = \sum_{e \in E} tf(e, E)p(e|f) \quad \text{and} \quad df(f) = \sum_{e \in E} df(e)p(e|f)
  \]

- estimate \(p(e|f)\)'s from
  - lexical translation table
  - word alignments in derivations of the SMT \(n\)-best list

(2) Retrieval

- Okapi BM25 ranking
Section 2

Learning Phrase-Tables from Ranking Data
Ranking Approach: Model

- query \( q \), document \( d \) (bag-of-words)

**Scoring function**

\[
f(q, d) = q^\top W d = \sum_{i=1}^{Q} \sum_{j=1}^{D} q_i W_{ij} d_j.
\]

**Linear model**

Assign a weight to every pair of query and document terms:

\[
f(q, d) = \sum_{ij} W_{ij} (qd^\top)_{ij} = w^\top \phi(q, d)
\]
Training

Training data

\[ \{(q, d^+, d^-)\} \]

where \( d^+ \) is a relevant document and \( d^- \) an irrelevant for query \( q \)

Task

Find \( W \in \mathbb{R}^{Q \times D} \) such that \( f(q, d^+) > f(q, d^-) \) for all training tuples

How to learn big \( W \)?

- low-rank decomposition of \( W \) [Bai 10]
- force feature selection by \( \ell_1 \)-regularization [Chen 10]
- start from empty \( W \) & add features progressively
Training

Training data

\[(q, d^+, d^-)\]

where \(d^+\) is a relevant document and \(d^-\) an irrelevant for query \(q\)

Task

Find \(W \in \mathbb{R}^{Q \times D}\) such that \(f(q, d^+) > f(q, d^-)\) for all training tuples

How to learn big \(W\)?

- low-rank decomposition of \(W\) [Bai 10]
- force feature selection by \(\ell_1\)-regularization [Chen 10]
- start from empty \(W\) & add features progressively
Boosting

Exp loss function

\[
\mathcal{L}_{exp} = \sum_{(q,d^+,d^-)} D(q,d^+,d^-) e^{f^T(q,d^-) - f^T(q,d^+)}
\]

\[
D(q,d^+,d^-) \text{ – importance weighting from relevance levels}
\]

Iterative building of scoring function

\[
f^T(q,d) = \sum_{t} w_{ij}^t q_i^t d_j^t
\]

- on step \( t \) selects new pair \( i, j \)
- \( D_{t+1} \) reweighted to concentrate on previously misclassified pairs
An Efficient Implementation of Boosting

Parallelization & Bagging
- each node receives a sample $s$ from training tuples
- when done models are averaged: $f(q, d) = \frac{1}{S} \sum_t \sum_s w_t^s h_t^s(q, d)$

Speed & Memory Tricks
- on-the-fly feature construction (avoids inv. index) [Grangier 08, Goel 08]
- only update gradients for features that cooccur with previously selected one [Collins 05]
- random feature hashing into $2^{30}$-sized pool (keep $W$ in RAM) [Shi 09]
An Efficient Implementation of Boosting

Parallelization & Bagging
- each node receives a sample $s$ from training tuples
- when done models are averaged: $f(q, d) = \frac{1}{S} \sum_t \sum_s w_t^s h_t^s(q, d)$

Speed & Memory Tricks
- on-the-fly feature construction (avoids inv. index) [Grangier 08, Goel 08]
- only update gradients for features that cooccur with previously selected one [Collins 05]
- random feature hashing into $2^{30}$-sized pool (keep $W$ in RAM) [Shi 09]
# Example Learned Phrase-Table

<table>
<thead>
<tr>
<th>t</th>
<th>$h_t$ (uni- &amp; bi-grams)</th>
<th>$w_t$</th>
</tr>
</thead>
<tbody>
<tr>
<td>1</td>
<td>層 layer - layer</td>
<td>1.29</td>
</tr>
<tr>
<td>2</td>
<td>データ data - data</td>
<td>1.13</td>
</tr>
<tr>
<td>3</td>
<td>回路 circuit - circuit</td>
<td>1.13</td>
</tr>
<tr>
<td>77</td>
<td>導 guide, 電 power - conductive</td>
<td>1.25</td>
</tr>
<tr>
<td>81</td>
<td>解決 resolution - image</td>
<td>-0.25</td>
</tr>
<tr>
<td>99</td>
<td>変速 speed - transmission</td>
<td>1.68</td>
</tr>
<tr>
<td>100</td>
<td>液晶 LCD - liquid, crystal</td>
<td>1.73</td>
</tr>
<tr>
<td>123</td>
<td>力 power - force</td>
<td>0.91</td>
</tr>
<tr>
<td>124</td>
<td>圧縮 compression, 機 machine - compressor</td>
<td>2.83</td>
</tr>
<tr>
<td>132</td>
<td>ケーブル cable - cable</td>
<td>1.81</td>
</tr>
<tr>
<td>133</td>
<td>超 hyper, 音波 sound wave - ultrasonic</td>
<td>3.34</td>
</tr>
<tr>
<td>169</td>
<td>粒子 particle - particles</td>
<td>1.57</td>
</tr>
<tr>
<td>170</td>
<td>算出 calculation - for, each</td>
<td>1.14</td>
</tr>
<tr>
<td>184</td>
<td>ロータ rotor - rotor</td>
<td>2.01</td>
</tr>
<tr>
<td>185</td>
<td>検出 detection, 器 vessel - detector</td>
<td>1.43</td>
</tr>
</tbody>
</table>
Section 3

Experiments
## Parallel Translation Data (JP→EN)

<table>
<thead>
<tr>
<th>Experiments</th>
<th>Setup</th>
</tr>
</thead>
<tbody>
<tr>
<td><strong>Training</strong></td>
<td></td>
</tr>
<tr>
<td>NTCIR-7 PatentMT workshop data (1.8M sentences)</td>
<td></td>
</tr>
<tr>
<td><strong>Parameter tuning</strong></td>
<td></td>
</tr>
<tr>
<td>parameter tuning: NTCIR-8 test collection (2K sentences)</td>
<td></td>
</tr>
</tbody>
</table>
**Ranking Data**

### Automatic extraction of relevance judgements [Graf 08]

- Cross-language citation graph from MAREC corpus to extract patents in citation or family relation
- 3 relevance levels:
  - 3 family patents (same invention granted elsewhere)
  - 2 cited by examiners
  - 1 cited by applicants

- Extracted abstracts from MAREC and NTCIR-10

<table>
<thead>
<tr>
<th>queries</th>
<th>relevant docs</th>
</tr>
</thead>
<tbody>
<tr>
<td>train</td>
<td>100k</td>
</tr>
<tr>
<td>dev</td>
<td>2k</td>
</tr>
<tr>
<td>test</td>
<td>2k</td>
</tr>
</tbody>
</table>

http://www.cl.uni-heidelberg.de/statnlpgroup/boostclir
Performance of standalone systems

- MAP - Mean Average Precision
- PRES - Patent Retrieval Evaluation Score (recall-oriented) [Magdy 11]
- both $\in [0, 1]$; higher is better

<table>
<thead>
<tr>
<th></th>
<th>test MAP</th>
<th>test PRES</th>
</tr>
</thead>
<tbody>
<tr>
<td>DT$^1$</td>
<td>0.2555</td>
<td>0.5681</td>
</tr>
<tr>
<td>PSQ lexical table$^2$</td>
<td>0.2444</td>
<td>0.5498</td>
</tr>
<tr>
<td>PSQ $n$-best table$^3$</td>
<td>0.2659</td>
<td>0.5851</td>
</tr>
<tr>
<td>Boost-unigram</td>
<td>$^{1,2,3}0.1982$</td>
<td>$^{1,2}0.6122$</td>
</tr>
<tr>
<td>Boost-bigram</td>
<td>$^30.2474$</td>
<td>$^{1,2,3}0.7196$</td>
</tr>
</tbody>
</table>

- small boosting models: $\sim100K$ (1-gram) & $\sim170K$ (2-gram)
- lexical table: $\sim600K$ entries
Rank Aggregation

Intuition

- SMT helpful for cohesive, general passages
- Boosting provides task-specific info complementary to SMT:
  - **✓** rewards phrase pairs that aid retrieval and
  - **✓** penalizes pairs that are detrimental to the task

Best of both worlds

- aggregate systems with orthogonal information sources
- consensus voting (Borda Count) + interpolation:

\[
 f_{agg}(q, d) = \kappa \frac{f_1(q, d)}{\sum_d f_1(q, d)} + (1 - \kappa) \frac{f_2(q, d)}{\sum_d f_2(q, d)}
\]
Rank Aggregation

Intuition

- SMT helpful for cohesive, general passages
- Boosting provides task-specific info complementary to SMT:
  - ✔️ rewards phrase pairs that aid retrieval and
  - ✔️ penalizes pairs that are detrimental to the task

Best of both worlds

- aggregate systems with orthogonal information sources
- consensus voting (Borda Count) + interpolation:

\[ f_{agg}(q, d) = \kappa \frac{f_1(q, d)}{\sum_d f_1(q, d)} + (1 - \kappa) \frac{f_2(q, d)}{\sum_d f_2(q, d)} \]
Performance of aggregated systems: MAP

dev set

DT + Boost-2g
PSQ lexical + Boost-2g
PSQ n-best + Boost-2g
DT, 0.2636
PSQ lexical, 0.2520
PSQ n-best, 0.2698
Boost-2g, 0.2526
PSQ n-best + DT

A. Sokolov, L. Jehl, F. Hieber, S. Riezler
Boosting CLIR by Learning Bilingual Phrase Associations
Performance of aggregated systems: PRES

dev set

DT + Boost-2g
PSQ lexical + Boost-2g
PSQ n-best + Boost-2g
DT, 0.5669
PSQ lexical, 0.5445
PSQ n-best, 0.5789
Boost-2g, 0.6900
PSQ n-best + DT
## Performance aggregated systems: overall

<table>
<thead>
<tr>
<th>method</th>
<th>test MAP</th>
<th>test PRES</th>
</tr>
</thead>
<tbody>
<tr>
<td>DT + PSQ n-best</td>
<td>*0.2726</td>
<td>*0.5942</td>
</tr>
<tr>
<td>DT + Boost-1g</td>
<td>*0.2728</td>
<td>*0.6225</td>
</tr>
<tr>
<td>DT + Boost-2g</td>
<td>*0.3300</td>
<td>*0.7279</td>
</tr>
<tr>
<td>PSQ lexical + Boost-1g</td>
<td>*0.2653</td>
<td>*0.6131</td>
</tr>
<tr>
<td>PSQ lexical + Boost-2g</td>
<td>*0.3187</td>
<td>*0.7240</td>
</tr>
<tr>
<td>PSQ n-best + Boost-1g</td>
<td>*0.2850</td>
<td>*0.6402</td>
</tr>
<tr>
<td>PSQ n-best + Boost-2g</td>
<td>*0.3416</td>
<td>*0.7376</td>
</tr>
</tbody>
</table>

- aggregating two SMT-based systems does not help!
- aggregating orthogonal systems gives up to +7 MAP/+15 PRES
Take-away message

- encode task-relevant information into “phrase-table”
- orthogonal & complementary information to standard CLIR
- aggregation with standard SMT gives a huge boost in performance

Data available at

http://www.cl.uni-heidelberg.de/statnlpgroup/boostclir
Take-away message

- encode task-relevant information into “phrase-table”
- orthogonal & complementary information to standard CLIR
- aggregation with standard SMT gives a huge boost in performance

Data available at

http://www.cl.uni-heidelberg.de/statnlpgroup/boostclir

Thank you
Experiments

Bing Bai, Jason Weston, David Grangier, Ronan Collobert, Kunihiko Sadamasa, Yanjun Qi, Olivier Chapelle & Kilian Weinberger.

*Learning to Rank with (a Lot of) Word Features.*

Xi Chen, Bing Bai, Yanjun Qi, Qihang Ling & Jaime Carbonell.

*Learning Preferences with Millions of Parameters by Enforcing Sparsity.*
In Proceedings of the IEEE International Conference on Data Mining (ICDM’10), Sydney, Australia, 2010.

Michael Collins & Terry Koo.

*Discriminative Reranking for Natural Language Parsing.*

Kareem Darwish & Douglas W. Oard.

*Probabilistic Structured Query Methods.*


Sharad Goel, John Langford & Alexander L. Strehl.

*Predictive Indexing for Fast Search.*

Erik Graf & Leif Azzopardi.

*A methodology for building a patent test collection for prior art search.*

David Grangier & Samy Bengio.

*A discriminative kernel-based approach to rank images from text queries.*

Results

A. Sokolov, L. Jehl, F. Hieber, S. Riezler

Boosting CLIR by Learning Bilingual Phrase Associations
<table>
<thead>
<tr>
<th>Experiments</th>
<th>Results</th>
</tr>
</thead>
</table>
| Kenneth Heafield.  
*KenLM: Faster and Smaller Language Model Queries.*  
| Walid Magdy & Gareth J. F. Jones.  
| Franz Josef Och.  
*Minimum error rate training in statistical machine translation.*  
In Proceedings of the 41st Meeting on Association for Computational Linguistics (ACL’03), Sapporo, Japan, 2003. | |
*Hash Kernels.*  
In Proceedings of the 12th Int. Conference on Artificial Intelligence and Statistics (AISTATS’09), Irvine, CA, 2009. | |
| Ferhan Ture, Jimmy Lin & Douglas W. Oard.  
*Looking Inside the Box: Context-Sensitive Translation for Cross-Language Information Retrieval.*  
Verification that gains transfer to the test data.

A. Sokolov, L. Jehl, F. Hieber, S. Riezler

Boosting CLIR by Learning Bilingual Phrase Associations