Raytheon
BBN Technologies

Learning to Translate: A Query-Specific Combination Approach for Cross-Lingual Information Retrieval (CLIR)

Ferhan Ture and Elizabeth Boschee
{fture,eboschee}@bbn.com

Overview

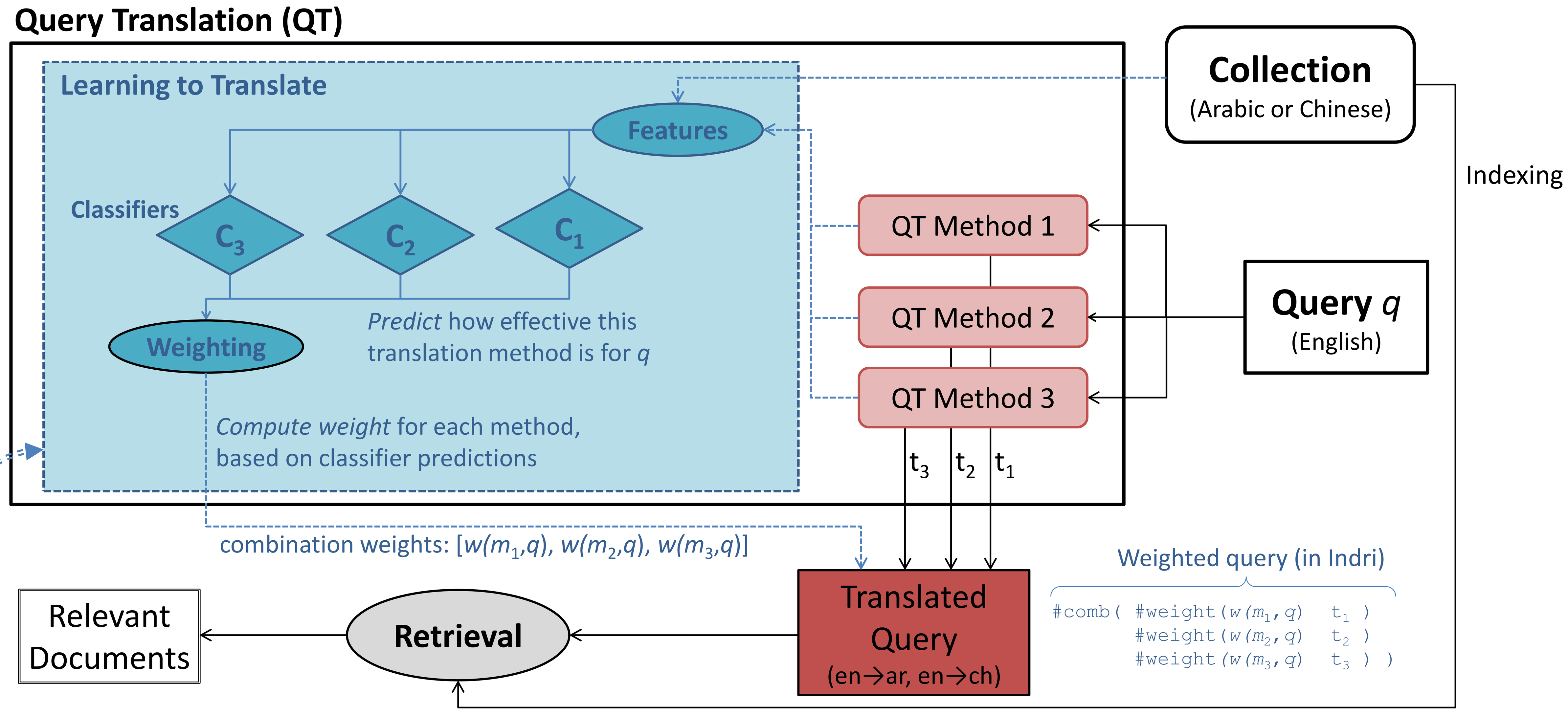
Cross-Lingual Information Retrieval (CLIR)
Retrieve documents in a collection that are relevant to a query, where the query and documents are in different languages.

- Typically, the query is translated into the document language, or vice versa.
- Previous work has shown that:

1. Combining different translation methods is beneficial.
2. Effectiveness of each method differs greatly across queries and tasks.

➔ **Our approach**: A novel classification-based approach for learning how much weight to put on each translation method, for each query.

Query Translation (QT)



Query Translation (QT)

Baseline

Methods
1-best: Output of Machine Translation (MT) system
***N*-best**: Top *N* translations converted into a word by word probability distribution.
Word-based: Translation probabilities induced from word alignments from a parallel corpus.

Each method has strengths and weaknesses
→ We take advantage of this fact to improve overall CLIR effectiveness.

Uniform: Naively assign equal weights to each method.

Task-Specific: For every query, use combination weights that maximize retrieval effectiveness on a tuning task.

Query-Specific: Compute combination weights *specifically for each query*; takes into account how effectiveness of each QT method varies across queries.

Our Approach

Learning to Translate

A binary classification problem is designed for each translation method *m*:

Each query *q* is converted into a training instance using one of the two labeling methods:

Labeling
By-measure Label=1 if *m* performs at least 90% as well as the best method.
By-rank Sort queries based on *m*'s effectiveness; top half gets Label=1.

We discard queries for which there is *negligible* difference between the effectiveness of the best and worst translation method.

Features

Surface	*Parse*	*Translation*	*Index*
• Length • # stop words • Type of question	• Named entity • Syntactic constituents (e.g., is there an adverb in the query?)	• # unaligned words • # multi-aligned words • # self-aligned words • Entropy of translation probability distribution	• Term frequency (tf) • Document frequency (df) • Probability assigned to OOV words

We train MaxEnt, SVM and decision tree classifiers using various feature subsets.
The best classifier is selected on *tuning* data, under one of the three scenarios:

Tuning

Fully-open	*Half-blind*	*Fully-blind*
Train/tune in-domain "leave one out"	Train out-domain Tune in-domain	Train out-domain Tune out-domain

→ After training, classifier C_m is used for determining the weight of *m* in retrieval:

weight(m,q) ~ confidence of C_m that Label=1 for *q*

Evaluation

Baseline CLIR

Task	1-best	10-best	Word	Uniform	Task-Specific
	QT baselines			Combination baselines	
$BOLT_{ar}$	0.296	0.311	0.318	0.324[12]	**0.329**[1]
$BOLT_{ch}$	0.370	0.406	0.407	0.422[1]	**0.431**[1]
$TREC_{ar}$	0.292	0.298	0.301	0.314[1]	**0.318**[1]
$NTCIR_{ch}$	0.146	0.152	0.141	**0.162**[13]	**0.162**[13]

Superscripts [1 2 3] indicate statistically significant improvements ($p < 0.05$) over 1-best, 10-best, word-based approaches.

Mean Average Precision (MAP): measures how well the retrieval system ranked relevant documents w.r.t non-relevant ones.

Our Approach: Effect of Labeling

Task	By-Measure		By-Rank	
$BOLT_{ar}$	**0.342**[*]		0.330	
	74%	72%	72%	71%
$BOLT_{ch}$	**0.438**		0.426	
	68%	72%	60%	71%
$TREC_{ar}$	0.305		**0.316**	
	59%	82%	59%	65%
$NTCIR_{ch}$	**0.163**		0.162	
	56%	64%	61%	67%

Percentage of queries for which the classifier was correct.
Second value ignores "negligible queries".

Notes
To compare the two labeling approaches, we trained a classifier with each and scored the retrieved documents (MAP):
- For $BOLT_{ar}$, *by-measure* is stat. sig. better.
- For TREC, difference is due to two outlier queries (not stat. sig.).

→ We decided to use *by-measure* labeling.

Most Informative Features
• Translation-based and index-based features are selected in the best feature subset in almost all cases.
• Parse-based features most helpful in classifying NTCIR queries.

Our Approach: Effect of Tuning

Task	Query-Specific Combination			Oracle
	Open	Half	Blind	
$BOLT_{ar}$	**0.342**[*†] bolt	0.330 t+n	0.329 t+n	0.346
$BOLT_{ch}$	**0.438**[*†] bolt	0.428 ntcir	0.426 t+n	0.466
$TREC_{ar}$	0.321 b+t	**0.324**[*†] b+n	0.321 b+n	0.332
$NTCIR_{ch}$	**0.164**[*] b+n	0.163 b+t	0.163 bolt	0.182

Notes
- Our best MAP for each task is highlighted in boldface.
- "Oracle" is a hypothetical system that selects the best method for each query.
- Superscripts [* †] indicate stat. sig. improvements over uniform and task-specific approaches.

Classifiers trained on BOLT and TREC queries.
Tuned on held-out portion of NTCIR, tested on remaining.

Conclusions

- Our results emphasize the benefit of query combination in CLIR, as it outperforms any single QT method on a variety of CLIR tasks.
- Finding a *custom combination recipe* for each query improves results even further.
- Degree of success depends highly on the availability of in-domain training/tuning data.

Future Work

- Our simple binary classifiers were successful on all four tasks, but we would like to experiment with a learning-to-rank framework for better optimization.
- Extending this approach to document translation is another interesting future direction.