Learning to Translate for Multilingual Question Answering

(Blinded for reviews)

ABSTRACT

We introduce a machine learning-based approach for optimizing translation in multilingual question answering (MLQA). In MLQA, there are multiple methods to translate the question and/or answers, four of which we explore in this paper. We build a feature for each combination of translation *direction* and *method*, and train a model that learns optimal feature weights. On a large multilingual forum, our novel *learn-to-translate* approach was more effective than a typical MLQA approach ($p < 0.05$): translating all text into English, then training a classifier based only on English (original or translated) text.

1. INTRODUCTION

Question answering (QA), the task of finding relevant well-formed answers to a posed question, usually consists of three main stages: (a) preprocessing the question and collection, (b) retrieval of candidate answers in the collection, and (c) ranking answers with respect to their relevance to the question and return the top N answers. The questions can range from factoid (e.g., "What is the capital of France?") to causal (e.g., "Why are trees green?"), and opinion questions (e.g., "What do people think about lowering the drinking age in the United States?"). While earlier research has mostly focused on relatively simpler factoid questions, in which there is a single clear answer that can potentially be retrieved from a structured database, this is no longer a representative case in many real-world applications.

The most common approach to *multilingual QA* (MLQA) has been to translate all content into its most probable English translation via machine translation (MT) systems. This strong baseline, which we refer to as *one-best MT* (`1MT`), has been successful in prior work [1, 9, 13, 11, 24]. However, recent advances in cross-lingual IR (CLIR) show that one can do better by representing the translation space as a probability distribution [25]. In addition, MT systems perform substantially worse with user-generated text, such as web forums [26], which provides extra motivation to consider alternative translation approaches for higher recall. To our knowledge, it has yet to be shown whether these recent advancements in CLIR transfer to MLQA.

We introduce a novel approach for MLQA, referred to as *Learning to Translate* (`L2T`), by developing a model that weights multiple translations of the question and/or candidate answer, based on how well it discriminates between good and bad answers. Each translation of the question and/or answer is represented by a feature (Section 2.1). The model then learns feature weights for each combination of translation *direction* and *method*, through a discriminative training process (Section 2.2). In addition to our novel features, we also experimented with various data selection strategies to optimize model training (Section 2.3). Experiments were conducted on the DARPA Broad Operational Language Technologies (BOLT) IR task.[1] Results confirm that our `L2T` approach yields statistically significant improvements `1MT` ($p < 0.05$).

Related Work: Research in QA has mostly been driven by annual evaluation campaigns like TREC,CLEF and NTCIR. Most earlier work relied on either manually crafted rule-based approaches, or traditional IR-based approaches where each pair of question and candidate answer was scored using retrieval functions (e.g., BM25 [22]). Alternatively, training a classifier for ranking candidate answers allows the exploitation of various features extracted from the question, candidate answer, and surrounding context [12, 27]. In fact, an explicit comparison at 2007 TREC confirmed the superiority of machine learning-based approaches (F-measure 35.9% vs 38.7%) [27]. Learning-to-rank approaches have also been applied to QA successfully [2].

When dealing with multilingual collections, most prior approaches translate all text into English beforehand, then treat the task as monolingual retrieval (previously referred to as `1MT`). At recent evaluation campaigns like CLEF and NTCIR,[2] almost all teams simply obtained the one-best question translation, treating some online MT system as a black box [1, 9, 13, 11, 24], with few notable exceptions that took term importance [21], or semantics [17] into account.

Contributions: Ture and Lin recently described three methods for translating queries into the collection language in a probabilistic manner, improving *document retrieval* effectiveness over a one-best translation approach [25]. Extending this idea to MLQA appears as a logical next step,

Permission to make digital or hard copies of all or part of this work for personal or classroom use is granted without fee provided that copies are not made or distributed for profit or commercial advantage and that copies bear this notice and the full citation on the first page. Copyrights for components of this work owned by others than ACM must be honored. Abstracting with credit is permitted. To copy otherwise, or republish, to post on servers or to redistribute to lists, requires prior specific permission and/or a fee. Request permissions from permissions@acm.org.

WOODSTOCK '97 El Paso, Texas USA

© 2015 ACM. ISBN 123-4567-24-567/08/06…$15.00

DOI: 10.475/123_4

[1] http://www.darpa.mil/Our_Work/I2O/Programs/Broad_Operational_Language_Translation_(BOLT).aspx

[2] Most recent MLQA track was in 2008 for CLEF, and 2010 for NTCIR.

yet most prior work rely solely on the one-best translation of questions or answers [10, 8, 4], or select the best translation out of few options [23, 16]. Mehdad et al. reported improvements by using a distance-based entailment score to choose among the top ten translations [14]. To the best of our knowledge, there is no prior work in the literature, where the ***optimal query and/or answer translation is learned via machine learning***. In addition to learning the optimal translation, we show that ***learning the optimal subset of the training data for a given task*** improves effectiveness. We select data based on either the source language of the sentence, or the annotation language. Such data selection strategies have not been studied extensively in the QA literature, therefore our results can provide useful insights to the community. With these two contributions, we outperform the state of the art.

2. APPROACH

Our work is focused on the *answer ranking* stage of QA: Given a natural-language question q in English and k candidate answers $s_1, \ldots, s_k$ (either English, Arabic, or Chinese), we score each answer in terms of its relevance to q. In our case, candidate answers are sentences extracted from all documents using the Indri retrieval engine [15]).

We aim to build a system that can successfully retrieve relevant information from open-domain and informal-language content. In this scenario, two assumptions made by many of the prior approaches fail: The assumptions i) that we can accurately classify questions via template patterns (Chaturvedi et al. argue that this does not hold for non-factoid questions [4]) and ii) that we can accurately determine the relevance of an answer, based on its automatic translation into English (Wees et al. show how recall decreases when translating user-generated text [26])

Instead, we opted for a more adaptable approach, in which question-answer relevance is modeled using a discriminative classifier that represents a function of features intended to capture the relationship between the question and sentence text. Also, instead of relying solely on a single potentially incorrect English translation, we increase our chances of a hit by translating both the question and the candidate answer, using four different translation methods.

Our main features, which are described throughout this section, are based on lexical similarity computed using these translations. The classifier is trained on a large number of question-answer pairs, each labeled by a human annotator with a binary relevance label.[3] We also show that restricting the classifier to a specific subset of training data can result in a much better model of relevance.

2.1 Representation

In MLQA, since questions and answers are in different languages, most approaches translate both into an intermediary language (usually English). As a result, valuable information often gets "lost in translation", due to the error-prone nature of MT. These errors are especially noticeable when translating informal text [26], or less-studied languages.

Translation Direction: In our approach, we perform a *two-way translation* to better retain the original meaning: in addition to translating each non-English sentence into English, we also translate the English questions into Arabic and Chinese (using multiple translation methods, described below). For a given question-answer pair, we have two"views": comparing the translated question to the original sentence (call this *collection-language* (CL) view); and comparing the original question to the translated sentence (call this *question-language* (QL) view).

Translation Method: When translating text for retrieval tasks like QA, including a variety of alternative translations is as important as finding the most accurate translation, especially for non-factoid questions, where capturing (potentially multiple) underlying topics is essential. We explored four *translation methods* for translating the English question into Arabic and Chinese. Each method outputs a probability distribution for each question word, expressing the translation space in the collection language:

Word: In MT, a word alignment is a many-to-many mapping between source- and target-language words, learned without supervision, at the beginning of the training pipeline [19]. These alignments can be converted into word translation probabilities [5]. For example, in an English-Arabic parallel corpus, if an English word appears m times in total and is aligned to a certain Arabic word k times, we assign a probability of $\frac{k}{m}$ for this translation. This simple idea has performed greatly in IR for generating a probability distribution for query word translations.

Grammar: Probabilities can be derived from a synchronous context-free grammar, which is a typical translation model found in MT systems [25]. The grammar contains rules r that follow the form $\alpha \;||\; \beta \;||\; \mathcal{A} \;||\; \ell(r)$, stating that source-language word α can be translated into target-language word β, with an associated likelihood value $\ell(r)$. $\mathcal{A}$ represents the word alignments. For each rule r that applies to the question text, we identify each source word s_j. From the word alignment information included in the rule, we can find all target words that s_j is aligned to. By processing all the rules to accumulate likelihood values, we can construct translation probabilities for each word in the question.

10-best: Statistical MT systems can output a ranked list of translations, instead of the single best. Ture and Lin exploited this to obtain word translation probabilities from the top 10 translations of the question [25].

For each question word w, we can extract which grammar rules were used to produce the translation – once we have the rules, word alignments allow us to find all target-language words that w translates into. By doing this for each question translation, we construct a probability distribution that defines the translation space of each question word.

Context: Neural network-based MT models learn context-dependent word translation probabilities – the probability of a target word is dependent on the source word it aligns to, as well as a 5-word window of context [6].

We modify the context-dependent lexical translation model from [6] for question translation. However questions are sometimes not part of a full, well-formed sentence, so we randomly replace words in the window with a special filler token. This teaches the model how to accurately translate with full context, partial context, and no context.

For example, the question "Tell me about child labor in Africa", which is simplified by our preprocessing engine to "child labor in Africa", is translated into the following probabilistic structure.

We are not aware of any other MLQA approach that rep-

[3]Annotators score each answer from 1 to 5. We label any score of 3 or higher as relevant.

Figure 1: Probabilistic grammar-based translation of example question q (call this q_{grammar}).

resents the question-answer pair based on their probabilistic translation space.

2.2 Features

Given two different translation directions (*CL* and *QL*), and four different translation methods (*Word*, *Grammar*, *10-best*, *Context*), our strategy is to leverage a machine learning process to determine how helpful each signal is with respect to the end task. For this, we introduced separate question-answer similarity features based on each combination of translation direction and method.

Figure 2 shows how the probabilistic structure in Figure 1 is converted into a single real-valued vector $v_{q_{\text{grammar}}}$ by averaging values for each Chinese word across the three distributions. Similarly, a candidate answer s in Chinese is represented by scoring each word by its frequency. Given the two vectors, we compute the cosine similarity. The same process is repeated for the other three translation methods. These four lexical collection-language similarity features are collectively called *LexCL*.

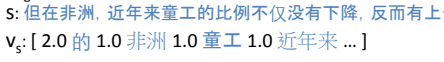

Figure 2: Vector representation of grammar-translated question (q_{grammar}) and sentence (s).

As mentioned before, we also obtain a similarity value by translating the sentence ($s_{1\text{best}}$) and computing the cosine similarity with the original question (q). Although it is possible to translate the sentence into English using the same four methods, we only used the one-best translation due to the computational cost. Hence, we have only one lexical similarity feature in the QL view (call *LexQL*). After computation, feature weights are learned via a maximum-entropy model.[4] We also include the same set of features from the previous sentence to represent the larger discourse.A wider context did not show further improvements in validation

2.3 Data Selection

There are at least two reasons why selecting training data based on language might benefit MLQA:
i) Separating translation quality from relevance: If translation has errors, relevant answers might be judged as non-relevant. Training on this data might lead to a tendency to favor English answers higher than Arabic or Chinese.
ii) Filtering out noisy labels: Since some pairs were annotated in both original language and English translation, independently, we can remove inconsistent ones from training.

In order to explore further, we generated seven different subsets of the training set by filtering instances with respect to (i) the original *language* of the answer, or (ii) the language of *annotation* (i.e., based on original text or its English translation):
lang=en: Sentences from the English corpus.
lang=ar/ch: Sentences from the Arabic / Chinese corpus (regardless of how it was judged).
annot=consist: All sentences except those that were judged inconsistently.
annot=src+: Sentences judged only in original text, or judged in both consistently.
annot=en+: Sentences that are judged either only in English, or judged in both original and English translation consistently.
all: All sentences.

[4]We also tried support vector machines and noticed worse results.

3. EVALUATION

In this section, we describe the evaluation of our multilingual QA approach. In order to perform controlled experiments and gain more insights, we split our evaluation into four separate tasks: three tasks focus on retrieving answers from posts written in a specified language (*English-only*, *Arabic-only*, or *Chinese-only*), and the last task is not restricted to any language (*Mixed-language*).

All experiments were conducted on the IR evaluation task of the DARPA BOLT program. The collection consists of 12.6m Arabic, 7.5m Chinese, and 9.6m English Web forum posts. All experimental runs use a set of 45 non-factoid (mostly opinion and causal) English questions, from a range of topics. All questions and forum posts were processed with an information extraction (IE) toolkit [reference removed for anonymization], which performs sentence-splitting, named entity recognition, coreference resolution, parsing, and part-of-speech tagging.

All non-English posts were translated into English, and all questions were translated into Arabic and Chinese, using state-of-the-art English↔Arabic (En-Ar) and English↔Chinese (En-Ch) MT systems [reference removed for anonymization]. Underlying models were trained on parallel corpora from NIST OpenMT 2012, in addition to parallel forum data collected as part of the BOLT program (10m En-Ar words; 30m En-Ch words). From these data, word alignments were learned with GIZA++ [20] (five iterations of IBM Models 1–4 and HMM). While we only used the one-best English translation for sentences, we applied the probabilistic translation methods from Section 2.1 to questions.

After all preprocessing, features were computed using the original post and question text, and their translations. Training data were created by having annotators label all sentences of the top 200 documents retrieved by Indri from each collection (for each question). Due to the nature of retrieval tasks, labels of the training data are usually unbalanced, with more negatively labeled sentences. In order to correct this, we split the data into balanced subsets (each sharing the same set of positively labeled data) and train multiple classifiers, then take a majority vote when predicting.

For testing, we froze the set of candidate answers and applied a trained classifier to each question-answer pair, generating a ranked list of answers for each question. Evaluation was performed on this ranked list by computing average precision (AP). Due to the size and redundancy of the collections, we sometimes end up with over 1000 known relevant answers for a question. So it is neither reasonable nor meaningful to compute AP until we reach 100% recall (e.g., 11-point AP) for these cases. Instead, we computed AP-k, by accumulating precision values at every relevant answer

until we get k relevant answers.[5]

Baseline: As described earlier, the baseline system computes similarity between question text and the one-best translation of the candidate answer (we run the sentence through our state-of-the-art MT system). After translation, we compute similarity via scoring the match between the parse of the question text and the parse of the candidate answer, using our finely-tuned IE toolkit [reference removed for anonymization]. This results in three different similarity features: matching the tree node similarity, edge similarity, and full tree similarity. Feature weights are then learned by training this classifier discriminatively on the training data described above. This already performs competitively, outperforming the simpler baseline where we compute a single similarity score between question and translated text, and matching the performance of the recently published system by Chaturvedi et al on the BOLT evaluation [4]. Baseline MAP values are reported on the leftmost column of Table 1.

Data effect: In the baseline approach, we do not perform any data selection, and use all available data for training the classifier. In order to test our hypothesis that selecting a linguistically-motivated subset of the training data might help, we used 10-fold cross-validation to choose the optimal data set (among seven options described in Section 2.3). Results indicate that including English or Arabic sentences when training a classifier for Chinese-only QA is a bad idea, since effectiveness increases when restricted to Chinese sentences (`lang=ch`). On the other hand, for the remaining three tasks, the most effective training data set is `annot=en+consist`. These selections are consistent across all ten folds, and the difference is statistically significant for all but Arabic-only. The second column in Table 1 displays the MAP achieved when data selection is applied before training the baseline model.

Feature effect: In order to measure the impact of our novel features (Section 2.2), we trained classifiers using either *LexCL*, *LexQL*, or *both* feature sets. In these experiments, the data is fixed to the optimal subset found earlier. Results are summarized on left side of Table 1. Statistically significants improvements over *Baseline* and *Baseline+Data selection* are indicated with single and double underlining, respectively.

For Arabic-only QA, adding *LexQL* features yields greatest improvements over the baseline, while the same statement holds for *LexCL* features for the Chinese-only task. For the English-only and mixed-language tasks, the most significant increase in MAP is observed with all of our probabilistic bilingual features. For all but Arabic-only QA, the MAP is statistically significantly better ($p < 0.05$) than the baseline; for Chinese-only and mixed-language tasks, it also outperforms baseline plus data selection ($p < 0.05$).[6] All of this indicates the effectiveness of our bilingual features and probabilistic question translation, as well as our data selection strategy.

Understanding the contribution of each of the four *LexCL* features is also important. To gain insight, we trained a classifier using all *LexCL* features (using the optimal data subset learned earlier for each task), and then incrementally removed one of the features, and tested on the same task. This controlled experiment revealed that the *word* translation feature is most useful for Chinese-only QA (i.e., removing it produces largest drop in MAP, 0.6 points), whereas *context* translation appears to be most useful (by a slighter margin) in Arabic-only QA. In the former case, the diversity provided by word translation might be useful at increasing recall in retrieving Chinese answers. In retrieving Arabic answers, using context to disambiguate the translation might be useful at increasing precision. This result further emphasizes the importance of a customized translation approach for MLQA.

Task	L2T				
	Baseline	+Data		+Feats	
Arz	0.421	0.423	eng+	0.425	LexQL
Cmn	0.416	0.425	cmn-only	0.451	LexCL
Eng	0.637	0.657	eng+	0.660	all
Mixed	0.665	0.675	eng+	0.681	all

Table 1: MAP with 10-fold cross-validation for each task. We compare numbers to *baseline* scores; statistically significant increase are underlined ($p < 0.05$).

Furthermore, to test the effectiveness of the probabilistic translation approach (Section 2.1), we replaced all *LexCL* features with a single lexical similarity feature computed from the one-best question translation. This resulted in lower MAP: 0.427 to 0.423 for Arabic-only, and 0.451 to 0.425 for Chinese-only task ($p < 0.01$), supporting the hypothesis that *probabilistic translation is more effective than the widely-used one-best translation.* In fact, almost all gains in Chinese-only QA seems to be coming from the probabilistic translation.

To test the robustness of our approach, we let cross-validation select the best combination of (*data*, *feature*), mimicking a less controlled, real-world setting. In this case, the best MAP for the Arabic-only, Chinese-only, English-only, and Mixed-language tasks are 0.403, 0.448, 0.657, and 0.679, respectively. In all but Arabic-only, these are statistically significantly better ($p < 0.05$) than not tuning the feature set or training data (i.e., Baseline). This result provides support that our approach can be used for any MLQA task out of the box, and provide improvements.

4. CONCLUSIONS

Our experimental analysis makes a thorough case on how language-inspired data selection and feature engineering affect QA effectiveness, and provides empirical evidence to fully support three of our hypotheses. With additional lexical and semantic similarity features from two views (question-language and collection-language), as well as a carefully selected training set, our classifier improved answer ranking effectiveness significantly for Chinese-only, English-only, and mixed-language QA.

An even more comprehensive use of machine learning would be to learn word-level translation scores, instead of relying on translation probabilities from the bilingual dictionary. This would result in a fully customized translation approach. Similar approaches have appeared in learning-to-rank literature for monolingual IR [3], we are not aware of such an approach for multilingual retrieval.

Another extension of this work would be to apply the

[5] k was fixed to 20 in our evaluation, although we verified that conclusions do not change with k set to 50, 100, or 150.
[6] Note that bilingual features are not expected to help on the English-only task, and the improvements come solely from data selection.

probabilistic translation methods for translating answers into the question language (in addition to question translation). By doing this, we would be able to capture the semantics of each answer much better, since we have discussed that one-best translation discards a lot of potentially useful information. For this extension, the processing pipeline would need to handle the increased computational complexity.

5. REFERENCES

[1] S. Adafre and J. van Genabith. Dublin city university at qaclef 2008. In C. Peters, T. Deselaers, N. Ferro, J. Gonzalo, G. Jones, M. Kurimo, T. Mandl, A. Peñas, and V. Petras, editors, *Evaluating Systems for Multilingual and Multimodal Information Access*, volume 5706 of *Lecture Notes in Computer Science*, pages 353–360. Springer Berlin Heidelberg, 2009.

[2] A. Agarwal, H. Raghavan, K. Subbian, P. Melville, R. D. Lawrence, D. C. Gondek, and J. Fan. Learning to Rank for Robust Question Answering. In *Proceedings of CIKM '12*, pages 833–842, New York, NY, USA, 2012. ACM.

[3] M. Bendersky, D. Metzler, and W. B. Croft. Learning concept importance using a weighted dependence model. In *Proceedings of WSDM '10*, pages 31–40, New York, NY, USA, 2010. ACM.

[4] S. Chaturvedi, V. Castelli, R. Florian, R. M. Nallapati, and H. Raghavan. Joint Question Clustering and Relevance Prediction for Open Domain Non-factoid Question Answering. In *Proceedings of WWW '14*, pages 503–514, New York, NY, USA, 2014. ACM.

[5] K. Darwish and D. W. Oard. Probabilistic structured query methods. In *Proceedings of SIGIR '03*, 2003.

[6] J. Devlin, R. Zbib, Z. Huang, T. Lamar, R. M. Schwartz, and J. Makhoul. Fast and robust neural network joint models for statistical machine translation. In *Proceedings of ACL '14*, June 22-27, 2014, Baltimore, MD, USA, pages 1370–1380, 2014.

[7] M. Esplà-Gomis, F. Sánchez-Martínez, and M. L. Forcada. UAlacant: Using Online Machine Translation for Cross-lingual Textual Entailment. In *Proceedings of the SemEval '12*, pages 472–476, Stroudsburg, PA, USA, 2012.

[8] M. A. García-Cumbreras, F. Martínez-Santiago, and L. A. Ureña López. Architecture and Evaluation of BRUJA, a Multilingual Question Answering System. *Inf. Retr.*, 15(5):413–432, Oct. 2012.

[9] S. Hartrumpf, I. GlÃÂűckner, and J. Leveling. Efficient question answering with question decomposition and multiple answer streams. In C. Peters, T. Deselaers, N. Ferro, J. Gonzalo, G. Jones, M. Kurimo, T. Mandl, A. PeÃ±as, and V. Petras, editors, *Evaluating Systems for Multilingual and Multimodal Information Access*, volume 5706 of *Lecture Notes in Computer Science*, pages 421–428. Springer Berlin Heidelberg, 2009.

[10] J. Ko, L. Si, E. Nyberg, and T. Mitamura. Probabilistic Models for Answer-ranking in Multilingual Question-answering. *ACM Trans. Inf. Syst.*, 28(3):16:1—-16:37, July 2010.

[11] C.-J. Lin and Y.-M. Kuo. Description of the NTOU complex QA system. In *Proceedings of NTCIR-8 Workshop*, 2010.

[12] N. Madnani, J. Lin, and B. J. Dorr. TREC 2007 ciQA Task: University of Maryland. *Proceedings of TREC '07*, 2007.

[13] Ã. Martinez-Gonzalez, C. de Pablo-Sanchez, C. Polo-Bayo, M. Vicente-Diez, P. Martinez-Fernandez, and J. L. Martinez-Fernandez. The miracle team at the CLEF 2008 multilingual question answering track. In C. Peters, T. Deselaers, N. Ferro, J. Gonzalo, G. Jones, M. Kurimo, T. Mandl, A. PeÃ±as, and V. Petras, editors, *Evaluating Systems for Multilingual and Multimodal Information Access*, volume 5706 of *Lecture Notes in Computer Science*, pages 409–420. Springer Berlin Heidelberg, 2009.

[14] Y. Mehdad, M. Negri, and M. Federico. Towards Cross-lingual Textual Entailment. In *Proceedings of NAACL-HLT '10*, pages 321–324, Stroudsburg, PA, USA, 2010.

[15] D. Metzler and W. B. Croft. A Markov random field model for term dependencies. In *Proceedings of SIGIR '05*, pages 472–479, New York, NY, USA, 2005. ACM.

[16] T. Mitamura, M. Wang, H. Shima, and F. Lin. Keyword translation accuracy and cross-lingual question answering in chinese and japanese. In *Proceedings of the Workshop on Multilingual Question Answering*, MLQA '06, pages 31–38, Stroudsburg, PA, USA, 2006.

[17] R. Munoz-Terol, M. Puchol-Blasco, M. Pardino, J. M. Gomez, S. Roger, K. Vila, A. Ferrandez, J. Peral, and P. Martinez-Barco. Integrating logic forms and anaphora resolution in the aliqan system. In *Evaluating Systems for Multilingual and Multimodal Information Access*, LNCS, pages 438–441. Springer Berlin Heidelberg, 2009.

[18] J.-Y. Nie. Cross-language information retrieval. *Synthesis Lectures on Human Language Technologies*, 3(1):1–125, 2010.

[19] F. J. Och. Minimum error rate training in statistical machine translation. In *Proceedings of ACL '03*, pages 160–167, Stroudsburg, PA, USA, 2003.

[20] F. J. Och and H. Ney. A systematic comparison of various statistical alignment models. *Computational Linguistics*, 29(1):19–51, 2003.

[21] H. Ren, D. Ji, and J. Wan. Whu question answering system at ntcir-8 aclia task. In *Proceedings of NTCIR-8 Workshop*, 2010.

[22] S. Robertson, H. Zaragoza, and M. Taylor. Simple {BM25} extension to multiple weighted fields. In *Proceedings of CIKM '04*, pages 42–49, 2004.

[23] B. Sacaleanu, G. Neumann, and C. Spurk. Dfki-lt at qaclef 2008. In C. Peters and et al., editors, *CLEF 2008 Working Notes*, Working Notes. Springer Verlag, 2008.

[24] H. Shima and T. Mitamura. Bootstrap pattern learning for open-domain clqa. In *Proceedings of NTCIR-8 Workshop*, 2010.

[25] F. Ture and J. Lin. Exploiting representations from statistical machine translation for cross-language information retrieval. *ACM Trans. Inf. Syst.*, 32(4):19:1–19:32, Oct. 2014.

[26] M. Van der Wees, A. Bisazza, W. Weerkamp, and C. Monz. What's in a domain? analyzing genre and topic differences in statistical machine translation. In

Proceedings of ACL-IJCNLP '15 (Volume 2: Short Papers), pages 560–566, Beijing, China, July 2015.

[27] C. Zhang, M. Gerber, T. Baldwin, S. Emelander, J. Chai, and R. Jin. Michigan State University at the 2007 TREC ciQA Task. In *Proceedings of TREC '07*, Gaithersburg, Maryland, Nov. 2007.