Learning to Translate for Multilingual Question Answering

(Blinded for reviews)

1. INTRODUCTION

Question answering (QA) usually consists of three 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. 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, which have 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, 23]. 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 conducted on the DARPA BOLT IR task[1] confirm that our `L2T` approach is statistically significantly better than `1MT`.

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 [21]). 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, 23], with few notable exceptions that took term importance [20], 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, yet most prior work rely solely on one-best translation of questions or answers [10, 8, 4], or select the best translation out of few options [22, 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 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.

SIGIR '16 San Francisco, Californoa USA

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.

2. APPROACH

Our work is focused on the *answer ranking* stage of QA: Given a natural-language question q in English, we score each candidate answer (either English, Arabic, or Chinese),[3] in terms of its relevance to q.

We aim to build a system that can successfully retrieve relevant information from open-domain and informal-language content. In this scenario, two common assumptions fail: (i) we can accurately classify questions via template patterns (Chaturvedi et al. argue that this does not hold for non-factoid questions [4]), and (ii) we can accurately determine the relevance of an answer, based on its one-best 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 multiple aspects between the question and sentence text. We describe details throughout this section.

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: 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 each question-answer pair, we have two "views": comparing translated question to the original sentence (i.e., *collection-language* (CL) view); and comparing original question to the translated sentence (i.e., *question-language* (QL) view).

Translation Method: When translating text for retrieval tasks, 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.[4]

<u>*Word*</u>: A word alignment is a many-to-many mapping between source- and target-language words, learned without supervision, during the MT training pipeline [19], which can be converted into word translation probabilities [5].

<u>*Grammar*</u>: Probabilities can be derived from a synchronous context-free grammar, a typical translation model found in MT systems [25]. Grammar contains rules that show how source phrases are translated into target phrases, with corresponding likelihood values. By processing all the rules to accumulate likelihood values, we can construct translation probabilities for each word in the question.

<u>*10-best*</u>: 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].

<u>*Context*</u>: 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].

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 ($q_{\texttt{grammar}}$) by *grammar* translation.

child: [0.32 童工 0.25 小孩 0.21 孩子 0.15 儿童 ...]
child labor child children child

labor: [0.36 童工 0.26 劳工 0.17 劳动 0.13 劳动力 ...]
labor labor labor force

Africa: [0.89 非洲 0.02 非 0.02 发展 0.01 南非 ...]
Africa non- development of South Africa

We are not aware of any other MLQA approach that represents 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.

Following shows how the probabilistic structure of $q_{\texttt{grammar}}$ is converted into a single real-valued vector, by averaging values for each Chinese word across the three distributions. Similarly, a candidate answer in Chinese is represented by scoring each word by its frequency, and cosine similarity is computed between the two vectors $v_{q_{\texttt{grammar}}}$ and v_s.

$v_{q_{grammar}}$: [0.30 非洲 0.23 童工 0.08 小孩 0.09 劳工 ...]
s: 但在非洲，近年来童工的比例不仅没有下降，反而有上升的趋势。
v_s: [2.0 的 1.0 非洲 1.0 童工 1.0 近年来 ...]

This process is repeated for each of the four translation methods, generating four lexical collection-language similarity features called *LexCL*.

As mentioned before, we also obtain a similarity value by translating the sentence (s_{1best}) 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.[5] Hence, we have only one lexical similarity feature in the QL view (called *LexQL*). After computation, feature weights are learned via a maximum-entropy model.[6] We also include the same set of features from the previous sentence to represent the larger discourse.[7]

2.3 Data Selection

There are at least two reasons why selecting training data based on language might benefit MLQA: (i) 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, and (ii) Since some pairs were annotated in both original language and English translation, independently, we can remove inconsistent

[3] In our case, candidate answers are sentences extracted from all documents using the Indri retrieval engine [15].

[4] We omit details due to space restrictions. See referenced papers for more details.

[5] This decision was primarily due to the time restrictions in our deployed application. Otherwise, it is quite straightforward to include those features as well.

[6] Support vector machines yielded worse results.

[7] A wider context did not show further improvements.

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): Sentences from the English corpus (`lang=en`), sentences from the Arabic / Chinese corpus (`lang=ar/ch`), sentences that were judged consistently (`annot=consist`), sentences judged only in original text, or judged in both consistently (`annot=src+`), sentences judged only in English, or judged in both consistently (`annot=en+`), or all sentences.

3. EVALUATION

In order to perform controlled experiments and gain more insight, we split our evaluation into four separate tasks: retrieval of answers from posts written in a specified language (*English-only (Eng)*, *Arabic-only (Arz)*, or *Chinese-only (Cmn)*), and retrieval without any restriction (*Mixed-language*). All experiments were conducted on the DARPA BOLT IR task, on a collection of 12.6m Arabic, 7.5m Chinese, and 9.6m English Web forum posts, and a set of 45 non-factoid (mostly opinion and causal) English questions All non-English posts were translated into English, and all questions were translated into Arabic and Chinese, using state-of-the-art Eng-Arz and Eng-Cmn MT systems [6]. The translation models were trained on parallel corpora from NIST OpenMT 2012, in addition to parallel forum data collected as part of the BOLT program (10m Eng-Arz words; 30m Eng-Cmn words). From these data, word alignments were learned with GIZA++ (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 per-question by Indri.

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.[8]

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.

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: Statistically significant increase over *baseline* and *+Data* are underlined (MAP with 10-fold cross-validation).

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.[9] The second column in Table 1 displays the MAP achieved when data selection is applied before training the baseline model.

Feature effect: 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 holds for *LexCL* features in 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 than the baseline; for Chinese-only and mixed-language tasks, it also outperforms baseline plus data selection.[10] 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 increasing recall in re-

[8] k was fixed to 20 in our evaluation, although we verified that conclusions do not change with k set to 50, 100, or 150.

[9] All statistical significance tests are based on [24] ($p < 0.05$).

[10] Note that bilingual features are not expected to help on the English-only task, and the improvements come solely from data selection.

trieving 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.

Furthermore, to test the effectiveness of probabilistic translation (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 than not tuning the feature set or training data (i.e., Baseline). This result provides support that our approach can be used out of the box for a given MLQA task.

4. CONCLUSIONS

Our experimental analysis makes a strong case on how our novel approach (i.e., probabilistic translation-based features and language-inspired data selection) can improve QA effectiveness. 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, resulting in a fully customized translation approach. Unlike monolingual IR [3], we are not aware of such an approach for multilingual retrieval. Another extension would be to apply the probabilistic translation methods for translating answers into the question language (in addition to question translation). By doing this, we would capture the semantics of each answer much better, as one-best translation discards a lot of potentially useful information.

5. REFERENCES

[1] S. Adafre and J. van Genabith. Dublin city university at qaclef 2008. In volume 5706 of *LNCS*. 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, 2012.

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

[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*, pages 503–514, 2014.

[5] K. Darwish and D. W. Oard. Probabilistic structured query methods. In *Proceedings of SIGIR*, 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*, 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*, 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 volume 5706 of *LNCS*, pages 421–428. 2009.

[10] J. Ko, L. Si, E. Nyberg, and T. Mitamura. Probabilistic Models for Answer-ranking in Multilingual Question-answering. *ACM TOIS*, 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*, 2010.

[12] N. Madnani, J. Lin, and B. J. Dorr. TREC ciQA Task: University of Maryland. *Proceedings of TREC*, 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 volume 5706 of *LNCS*, pages 409–420. 2009.

[14] Y. Mehdad, M. Negri, and M. Federico. Towards Cross-lingual Textual Entailment. In *Proceedings of NAACL-HLT*, 2010.

[15] D. Metzler and W. B. Croft. A Markov random field model for term dependencies. In *Proceedings of SIGIR*, 2005.

[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*, 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 LNCS, pages 438–441. 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*, 2003.

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

[21] S. Robertson, H. Zaragoza, and M. Taylor. Simple {BM25} extension to multiple weighted fields. In *Proceedings of CIKM*, 2004.

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

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

[24] M. D. Smucker, J. Allan, and B. Carterette. A comparison of statistical significance tests for information retrieval evaluation In *Proceedings of CIKM*, 2007.

[25] F. Ture and J. Lin. Exploiting representations from statistical machine translation for cross-language information retrieval. *ACM TOIS*, 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*, 2015.

[27] C. Zhang, M. Gerber, T. Baldwin, S. Emelander, J. Chai, and R. Jin. Michigan State University at the TREC ciQA Task. In *Proceedings of TREC*, 2007.