MT Error Detection for Cross-Lingual Question Answering

Kristen Parton
Columbia University
New York, NY, USA
kristen@cs.columbia.edu

Kathleen McKeown
Columbia University
New York, NY, USA
kathy@cs.columbia.edu

Abstract

We present a novel algorithm for detecting errors in MT, specifically focusing on content words that are deleted during MT. We evaluate it in the context of cross-lingual question answering (CLQA), where we try to correct the detected errors by using a better (but slower) MT system to retranslate a limited number of sentences at query time. Using a query-dependent ranking heuristic enabled the system to direct scarce MT resources towards retranslating the sentences that were most likely to benefit CLQA. The error detection algorithm identified spuriously deleted content words with high precision. However, retranslation was not an effective approach for correcting them, which indicates the need for a more targeted approach to error correction in the future.

1 Introduction

Cross-lingual systems allow users to find information in languages they do not know, an increasingly important need in the modern global economy. In this paper, we focus on the special case of cross-lingual tasks with result translation, where system output must be translated back into the user's language. We refer to tasks such as these as *task-embedded machine translation*, since the performance of the system as a whole depends on both task performance and the quality of the machine translation (MT).

Consider the case of cross-lingual question answering (CLQA) with result translation: a user enters an English question, the corpus is Arabic, and the system must return answers in English. If the corpus is translated into English before answer extraction, an MT error may cause the system to miss a relevant sentence, leading to decreased recall. Boschee et al. (2010) describe six queries from a formal CLQA evaluation where none of the competing systems returned correct responses, due to poor translation. In one example, the answer extractor missed a relevant sentence because the name “Abu Hamza al-Muhajir” was translated as “Zarqawi's successor Issa.” However, even if answer extraction is done in Arabic, errorful translations of the correct answer can affect precision: if the user cannot understand the translated English sentence, the result will be perceived irrelevant. For instance, the user may not realize that the mistranslation “Alry$Awy” refers to Al-Rishawi.

Our goal was not to improve a specific CLQA system, but rather to find MT errors that are likely to impact CLQA and correct them. We introduce an error detection algorithm that focuses on several common types of MT errors that are likely to impact translation adequacy:

- content word deletion
- out-of-vocabulary (OOV) words
- named entity missed translations

The algorithm is language-independent and MT-system-independent, and generalizes prior work by detecting errors at the word level and detecting errors across multiple parts of speech.

We demonstrate the utility of our algorithm by applying it to CLQA at query time, and investigate using a higher-quality MT system to correct the errors. The CLQA system translates the full corpus, containing 119,879 text documents and 150 hours of speech, offline using a production MT system, which is able to translate quickly (5,000 words per minute) at the cost of lower quality translations. A research MT system has higher quality but is too slow to be practical for a large amount of data (at 2 words per minute,

Coling 2010: Poster Volume, pages 946–954,
Beijing, August 2010

it would take 170 days on 50 machines to translate the corpus). At query-time, we can call the research MT system to retranslate sentences, but due to time constraints, we can only retranslate k sentences (we set k=25). In order to choose the sentences to best improve CLQA performance, we rank potential sentences using a relevance model and a model of error importance.

Our results touch on three areas:

- Evaluation of our algorithm for detecting content word deletion shows that it is effective, accurately pinpointing errors 89% of the time (excluding annotator disagreements).
- Evaluation of the impact of re-ranking shows that it is crucial for directing scarce MT resources wisely as the higher-ranked sentences were more relevant.
- Although the research MT system was perceived to be significantly better than the production system, evaluation shows that it corrected the detected errors only 39% of the time. Furthermore, retranslation seems to have a negligible effect on relevance. These unexpected results indicate that, while we can identify errors, retranslation is not a good approach for correcting them. We discuss this finding and its implications in our conclusion.

2 Task-Embedded MT

A variety of cross-lingual applications use MT to enable users to find information in other languages: e.g., CLQA, cross-lingual information retrieval (CLIR), and cross-lingual image retrieval. However, cross-lingual applications such as these typically do not do result translation – for instance, an English-French CLIR system would take an English query and return French documents, assuming that result translation is a separate MT problem. Part of the reason for the separation between cross-lingual tasks and MT is that evaluating task performance on MT is often difficult. For example, for a multilingual summarization task combining English and machine translated English, Daumé and Marcu (2006) found that doing a pyramid annotation on MT was difficult due to the poor MT quality.

Assessing cross-lingual task performance without result translation is problematic, because in a real-world application, result translation would affect task performance. For instance, in English-Arabic CLIR, a poorly translated relevant Arabic document may appear to be irrelevant to an English speaker. Decoupling the cross-lingual application from the MT system also limits the opportunity for feedback between the application and the MT system. Ji and Grishman (2007) exploited a feedback loop between Chinese and English named entity (NE) tagging and Chinese-English NE translation to improve both NE extraction and NE translation.

In this paper, error detection is done at query time so that query context can be taken into account when determining which sentences to retranslate. We also use the task context to detect errors in translating NEs present in the query.

3 Related Work

There is extensive prior work in describing MT errors, but they usually involve post-hoc error analysis of specific MT systems (e.g., (Kirchhoff et al., 2007), (Vilar et al., 2006)) rather than online error detection. One exception is Hermjakob et al. (2008), who studied NE translation errors, and integrated an improved on-the-fly NE transliterator into an SMT system.

Content word deletion in MT has been studied from different perspectives. Li et al. (2008) and Menezes and Quirk (2008) explored ways of modeling (intentional) source-word deletion in MT and showed that it can improve BLEU score. Zhang et al. (2009) described how errors made during the word-alignment and phrase-extraction phases in training phrase-based SMT often lead to spurious insertions and deletions during translation decoding. This is a common error – Vilar et al. (2006) found that 22% of errors produced by their Chinese-English MT system were due to missing content words. Parton et al. (2009) did a post-hoc analysis on the cross-lingual 5W task and found that content word deletion accounted for 17-22% of the errors on that task.

Some work has been done in addressing MT errors for different cross-lingual tasks. Ji and

1) Source	kmA tHdv wzyr AldfAE AlAsrA}yly Ayhwd bArAk Al*y zAr mwqE Altfjyr AlAntHAry fy dymwnp fy wqt sAbq En Altfjyr AlAntHAry . . .
ProdMT	There also the Israeli Defense Minister Ehud Barak, who visited the site of the suicide bombing in Dimona earlier, the suicide bombing . . .
Ref.	Moreover, Israeli Defense Minister Ehud Barak, who visited the scene of the suicide bombing in Dimona earlier, spoke about the suicide bombing . . .
2) Source	. . . Akd Ely rgbp hrAry AlAstfAdp mn AltjArb AlAyrAnyp fy mwAjhp Alqwy AlmEtdyp.
ProdMT	. . . stressed the desire to test the Iranian Harare in the face of the invading forces.
Ref.	. . . stressed Harare's desire to benefit from the Iranian experience in the face of the forces of aggressors.

Table 1: Two examples of content word deletion during MT.

Grishman (2007) detected NE translation errors in the context of cross-lingual entity extraction, and used the task context to improve NE translation. Ma and McKeown (2009) investigated verb deletion in Chinese-English MT in the context of CLQA. They tested two SMT systems, and found deleted verbs in 4-7% of the translations. After using post-editing to correct the verb deletion, QA relevance increased for 7% of the sentences, showing that an error that may have little impact on translation metrics such as BLEU (Papineni et al., 2002) can have a significant impact on cross-lingual applications.

Our work generalizes Ma and McKeown (2009) by detecting content-word deletions and other MT errors rather than just verb deletions. We also relax the assumption that translation preserves part of speech (i.e., that verbs must translate into verbs), assuming only that a phrase containing a content word should be translated into a phrase containing a content word. Instead of post-editing, we use an improved MT system to retranslate sentences with detected errors.

Using retranslation to correct errors exploits the fact that some sentences are harder to translate than others. In a resource-constrained setting, it makes sense to apply a better MT system only to sentences for which the fast MT system has lower confidence. We do not know of other systems that do multi-pass translation, but it is an interesting area for further work.

4 MT Error Detection

Most MT systems try to balance translation fluency with adequacy, which refers to the amount of meaning expressed in the original that is also expressed in the translation. For task-embedded MT, errors in adequacy are more likely to have an impact on performance than errors in fluency. Many MT metrics (such as BLEU) treat all tokens equally, so deleting a verb is penalized the same as deleting a comma. In contrast, we focus on errors in translating **content words**, which are words with open-class parts of speech (POS), as they are more likely to impact adequacy. First we describe how MT deletion errors arise and how we can detect them, and finally we describe detection of other types of errors.

4.1 Deletion in MT

The simplest case of content word deletion is a complete deletion by the translation model – in other words, a token was not translated. We assume the MT system produces word or phrase alignments, so this case can be detected by checking for a null alignment. However, it is necessary to distinguish correct deletion from spurious deletion. Some content words do not need to be translated – for example the Arabic copular verb "kAn" ("to be") is often correctly deleted when translating into English.

A more subtle form of content word deletion occurs when a content word is translated as a non-content word. This can be detected by comparing the parts of speech of aligned words. Consider the production MT System (Prod. MT) example in Table 1: the verb "tHdv"[1] ("spoke") has been translated as the expletive "there."

Finally, another case of content word deletion occurs when a content word is translated as part of a larger MT phrase, but the content word is not translated. In the second example in Table 1, an Arabic phrase consisting of a noun and preposition is translated as just the preposition "to."

[1] Arabic examples in this paper are shown in Buckwalter transliteration (Buckwalter, 2002).

The latter two kinds of content word deletion are considered mistranslations rather than deletions by the translation model, since the deleted source-language token does produce one or more target-language tokens. However, from the perspective of a cross-lingual application, there was a deletion, since some content that was present in the original is not present in the translation.

4.2 Detecting Deleted Content Words

The deletion detection algorithm is motivated by the assumption that a source-language phrase containing one or more meaning-bearing words should produce a phrase with one or more meaning-bearing words in the translation. (Phrase refers to an n-gram rather than a syntactic phrase.) Note that this does not assume a one-to-one correspondence between content words – for example, translating the phrase "spoke loudly" as the single word "yelled" satisfies the assumption. This hypothesis favors precision over recall, since it may miss cases where two content words are incorrectly translated as a single content word (for instance, if "coffee table" is translated as "coffee").

The algorithm takes as input POS tags in both languages and word alignments produced by the MT system during translation. The exact definition of "content word" will depend upon the language and POS tagset. The system iterates over all content words in the source sentence, and, for each word, checks whether it is aligned to one or more content words in the target sentence. If it has no alignment, or is aligned to only function words, the system reports an error. This rule-based approach has poor precision because of content words that are correctly deleted. For example, in the sentence "I am going to watch TV," "am" and "going" are tagged as verbs, but may be translated as function words. To address this, frequent content words were heuristically filtered using source-language IDF (inverse-document frequency) over the QA corpus. The cut-off was tuned on a development set.

This algorithm is a lightweight, language-independent and MT-system-independent way to find errors in MT. The only requirement is that the MT system produce word or phrase alignments. This algorithm generalizes Ma and McKeown (2009) in several ways. First, it detects any deleted content words, rather than just verbs. The previous work only addresses complete deletions, where the deleted token has a null alignment, whereas this approach finds cases where content words are mistranslated as non-content words. Finally, this error detection algorithm is more fine-grained, since it is at the word level rather than the phrase level.

4.3 Additional Error Detection Heuristics

For the CLQA task, we extended our MT error detection algorithm to handle two additional types of MT errors, OOV words and NE mistranslations, and to rank the errors. The production MT system was explicitly set to not delete OOV words, so they were easy to detect as source-language words left in the target language. The CLIR system was used to find occurrences of query NEs in the corpus, and then word alignments were used to extract the corresponding translations. If the translations were not a fuzzy match to the query, then it was flagged as a possible NE translation error. For instance, in a query about al-Rishawi, the CLIR would return Arabic-language matches to the Arabic word Alry$Awy. If the aligned English translation was al-Ryshoui instead of al-Rishawi, it would be flagged as an error.

Even if the retranslation corrects the errors in MT, if the sentences are not relevant, they will have no impact on CLQA. To account for relevance, we implemented a bilingual bag-of-words matching model, and ranked sentences with more keyword matches to the query higher. Sentences with the same estimated relevance were further sorted by potential impact of the MT error on the task. Errors affecting NEs (either via source-language POS tagging or source-language NE recognition) were ranked highest, since our particular CLQA task is focused on NEs. The final output of the algorithm is a list of sentences with MT errors, ranked by relevance to the query and importance of the error.

5 Experimental Setup

We begin by describing the MT systems, which motivate the need for time-constrained MT. Then we describe the CLQA task and the baseline CLQA system, and finally how the error detection algorithm is used by the CLQA system.

5.1 MT Systems

Both the research and production MT systems used in our evaluation were based on Direct Translation Model 2 (Ittycheriah and Roukos, 2007), which uses a maximum entropy approach to extract minimal translation blocks (one-to-M phrases with optional variable slots) and train system parameters over a large number of source- and target-language features. The research system incorporates many additional syntactic features and does a deeper (and slower) beam search, both of which cause it to be much slower than the production system. In addition, the research MT system filters the training data to match the test data, as is customary in MT evaluations, whereas the production system must be able to handle a wide range of input data. Part of the reason for the slower running time is that the research system has to retrain; the advantage is that more test-specific training data can be used to tailor the MT system to the input.

Overall, the research MT system performs 4 BLEU points better than the production MT system on a standard MT evaluation test corpus, but at a great cost: the production MT handles 5,000 words per minute, while the research MT system handles 2 words per minute. Using 50 machines, the production MT system could translate the corpus in under 2 hours, whereas the research MT system would take 170 days. This vast difference succinctly captures the motivation behind the time-constrained retranslation step.

5.2 CLQA Task

The CLQA task was designed for the DARPA GALE (Global Autonomous Language Exploitation) project. The questions found are open-ended, non-factoid information needs. There are 22 question types, and each type has its own relevance guidelines. For instance, one type is "Describe the election campaign of [PERSON]," and a question could be about Barack Obama. Queries are in English, the corpus is in Arabic, and the system must output comprehensible English sentences that are relevant to the question.

The Arabic corpus was created for the evaluation and consists of four genres: formal text (72,677 documents), informal text (47,202 documents), formal speech (50 hours), and informal speech (80 hours). The speech data was story segmented and run through a speech recognition system before translation. We used 31 text queries developed by the Linguistic Data Consortium (LDC), and 39 speech queries developed by other researchers working on the CLQA task.

5.3 CLQA System

The baseline CLQA system translates the full corpus offline before running further processing on the translated sentences (parsing, NE recognition, information extraction, etc.) and indexing the corpus. At query-time, CLIR (implemented with Apache Lucene) returns documents relevant to the query, and the CLQA answer extraction system is run over the translated documents. The answer extraction system relies on target-language annotations, but any MT errors will propagate to target-language processing, and therefore affect answer extraction.

5.4 CLQA System with MT Error Detection

The error detection and retranslation module was added to the baseline system after CLIR, but before answer extraction. The inputs to the detection algorithm are the query and a list of ranked documents returned by CLIR. The detection algorithm has access to the indexed (bilingual) corpus, source- and target-language annotations (POS tagging and NE recognition), and MT word alignments. The error detection algorithm has two stages: first it runs over sentences in documents related to the query, and after it finds $2k$ sentences with errors (or exhausts the document list), it reranks the errors as described in section 4.3 and retranslates the top k=25 sentences. Then the merged set of original and retranslated relevant sentences are passed to the answer extraction module.

By doing retranslation before answer extraction, the algorithm has the potential to improve precision and recall. An improved translation of a relevant Arabic sentence is more likely to be selected by the answer extraction system and increase recall, as in Boschee et al. (2010), where answers were missed due to mistranslation. A better translation of a relevant sentence is also more likely to be perceived as relevant, as shown by Ma and McKeown (2009).

6 Evaluation

Amazon Mechanical Turk (AMT) was used to conduct a large-scale evaluation of the impact of error detection and retranslation on relevance. An intrinsic evaluation of the error detection was run on a subset of the sentences, since it required bilingual annotators.

6.1 Task-Based Evaluation

Each sentence was annotated in the production MT version and the research MT version. The annotators were first presented with template relevance guidelines and an example question, along with 3 – 4 example sentences and expected judgments. Then the actual question was presented to the annotator, along with 5 sentences (all from a single MT system). For each sentence, the annotators were first asked to judge perceived adequacy and then relevance.

The *perceived adequacy* rating was loosely based upon MT adequacy evaluations – in other words, annotators were told to ignore grammatical errors and focus on perceived meaning. However, since there were no reference translations, annotators were asked to rate how much of the sentence they believed they understood by selecting one of (All, More than half, About half, Less than half, and None).

Genre	# detected errors per sentence	# detected errors per 1,000 tokens
Newswire	0.16	56
Broadcast news	0.23	105
Broadcast conversation	0.14	84

Table 2: Number of errors detected across different genres.

The *relevance* rating was based on the template relevance guidelines, and annotators could select one of (Relevant, Maybe relevant, Not relevant, Can't tell due to bad translation and Can't tell due to other reason).

6.2 Amazon Mechanical Turk (AMT)

The evaluation was run on AMT, which has been extensively used in NLP and has been shown to have high correlation with expert annotators on many NLP tasks at a lower cost (Snow et al., 2008). It has also been used in MT evaluation (Callison-Burch, 2009), though that evaluation used reference translations.

For 70 queries, the top 25 ranked sentences in both the production and research MT versions were evaluated. Each sentence was judged for both relevance and perceived adequacy by 5 annotators, for a total of 35,000 individual judgments. As is standard, some of the judgments were filtered due to noise by using the percent of time that an annotator disagreed with all other annotators, and the relative time spent on a given annotation. The percent of sentences with majority agreement was 91% for relevance and 72% for perceived adequacy.

6.3 Intrinsic Evaluation

Annotators were presented with an Arabic sentence with a single token highlighted, and asked whether the token was a "content word" or not. Then annotators were asked to decide which of two translations (in random order) translated the highlighted Arabic word best, or whether they were equal. In total, 150 sentences were judged by annotators with knowledge of Arabic. For both questions, kappa agreement was moderate.

7 Results

Table 2 shows how many errors were found by the error detection algorithm for each genre. Not surprisingly, more errors are detected in the speech genres (84 and 105 errors per 1,000 tokens) than in formal text (56 errors per 1,000 tokens). We attribute the large difference between broadcast news and broadcast conversa-

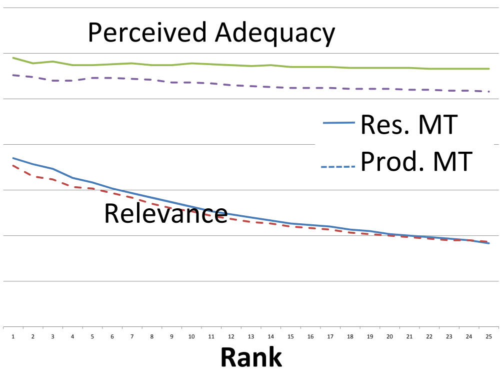

Figure 1: Average normalized cumulative sentence perceived adequacy and relevance versus rank of the sentence, by the ranking heuristic.

tion to the large number of short sentences without content words in informal speech (such as "hello", "thank you", etc.).

7.1 Perceived MT Adequacy

The research MT significantly outperformed the production MT in perceived adequacy (according to ANOVA with p=0.001). Of the production MT translations, 58% were considered "more than half" or "all" understandable, whereas 69% of the research MT were. Overall, retranslation increased perceived adequacy in 17% of the sentences, and decreased it in only 5% of sentences.

7.2 Ranking Algorithm

Figure 1 show the average cumulative sentence relevance and perceived adequacy, as ranked by the error detection algorithm. In other words, at each rank i, the average relevance (or perceived adequacy) of sentences $(1 - i)$ was calculated. On the perceived adequacy chart, the research MT system consistently outperforms the production MT system by a statistically significant margin. For relevance, the research MT curve is only marginally higher than the production MT curve.

The shape of the relevance curves shows that ranking sentences by a simple bilingual bag-of-words model did affect sentence relevance, since sentences that are higher ranked have higher cumulative average relevance. By ranking sentences with a basic relevance model, we were able to focus the scarce MT resources on sentences that are most likely to help the CLQA task. This underscores the importance of using the task context to guide MT error detection, especially in the case of time-constrained MT.

	Relevance				
	⇑	Same	⇓	No maj./ Don't know	
MT ⇑	20	201	9	56	17%
MT same	93	919	72	212	78%
MT ⇓	2	56	4	28	5%
	7%	70%	5%	18%	

Table 3: The relationship between changes in perceived adequacy and changes in relevance.

7.3 CLQA Relevance

Annotators judged 14.5% of the production MT sentences relevant. After retranslation, the overall number of sentences considered relevant increased to 14.7%. Although the overall numbers are similar, the relevance of many individual sentences did change. Table 3 shows the results of comparing annotations on the original MT with annotations on the retranslated MT. Relevance was classified as ⇑ or ⇓ by comparing the majority judgment of the production MT to the research MT. Changes in MT were based on comparing the average rating of both versions, with a tolerance of 1.0.

Of the sentences with better perceived MT, 7% increased in relevance, and 3% decreased in relevance. When the retranslated sentence was considered worse, there was a 2% increased in relevance and a 4% decrease. In other words, when retranslation had a positive effect, it more often led to increased relevance. However, the impact of retranslation was mixed, and none of the changes was statistically significant.

7.4 Intrinsic Evaluation

While the extrinsic evaluation focused on the impact on CLQA relevance, the goal of the intrinsic evaluation was to measure the precision of the error detection algorithm, and whether retranslation addressed the detected errors.

Of the 82% of sentences where both judges agreed, 89% of the detected errors were considered content words. All of the OOV tokens were content words (except for one disagreement). Surprisingly, for the errors involving content words, 60% of the time both systems were judged the same with regard to the highlighted error. The research system was better 39% of the time, and the original was better only 1% of the time (excluding 26% disagreements).

8 Discussion

The CLQA evaluation was based on three hypotheses:

- That we could detect errors in MT with high precision.
- That retranslating errorful sentences with a much better MT system would correct the errors we detected.
- That correcting errors would cause some sentences to become relevant which were not previously relevant, as in (Ma and McKeown, 2009).

The intrinsic evaluation confirmed that we can identify content word deletions in MT with high precision, thus validating the first hypothesis. However, detecting the errors and retranslating them did not lead to large improvements in CLQA relevance – the impact of increased perceived adequacy on relevance was mixed and not significant. The intrinsic evaluation explains this negative result: even though the retranslated sentences were judged significantly better, the retranslation only corrected the detected error 39% of the time. In other words, the better research MT system was making many of the same mistakes as the production MT system, despite using syntactic features and a much deeper search space during decoding. Since the second hypothesis did not hold, we need to improve our error correction algorithm before we can tell whether the third hypothesis holds.

This result directly motivates the need for targeted error correction of MT. Automatic MT post-editing has been successfully used for selecting determiners (Knight and Chander, 1994), reinserting deleted verbs (Ma and McKeown, 2009), correcting NE translations (Parton et al., 2008), and lexical substitutions (Elming, 2006). Since Arabic and English word order differ significantly, straightforward re-insertion of the deleted words is not sufficient for error correction, so we are currently working on more sophisticated post-editing techniques.

9 Conclusions

We presented a novel online algorithm for detecting MT errors in the context of a question, and a heuristic for ranking MT errors by their potential impact on the CLQA task. The error detection algorithm focused on content word deletion, which has previously been shown to be a significant problem in SMT. The algorithm is generally applicable to any MT system that produces word or phrase alignments for its output and any language pair that can be POS-tagged, and it is more fine-grained and covers more types of errors than previous work. It was able to detect errors in Arabic-English MT across multiple text and speech genres, and the intrinsic evaluation showed that the large majority of tokens flagged as errors were indeed content words.

The large-scale CLQA evaluation confirmed that the slower research MT system was significantly better than the production MT system. Relevance judgments showed that the ranking component was crucial for directing scarce MT resources wisely, as the higher-ranked sentences were most likely to be relevant to the query, and therefore most likely to benefit the CLQA system by being retranslated.

Although we correctly identified MT errors, retranslating the sentences with the errors had a negligible effect on CLQA relevance. This unexpected result may be explained by the fact that only 39% of the errors were actually corrected by the research MT system, so re-translation was not a good approach for error correction. We are currently working on correcting content word deletion in MT via post-editing.

Acknowledgments The authors are grateful to Radu Florian, Salim Roukos, Vittorio Castelli, Dan Bikel and the whole GALE IBM team for providing the experimental testbed, including the CLQA and MT systems. This research was partially supported by DARPA grant HR0011-08-C-0110.

References

Boschee, Elizabeth, Marjorie Freedman, Roger Bock, John Graettinger, and Ralph Weischedel. 2010. Error analysis and future directions for distillation. In *GALE book (in preparation)*.

Buckwalter, Tim. 2002. Buckwalter arabic morphological analyzer. *Linguistic Data Consortium. (LDC2002L49)*.

Callison-Burch, Chris. 2009. Fast, cheap, and creative: evaluating translation quality using amazon's mechanical turk. In *EMNLP '09*, pages 286–295, Morristown, NJ, USA. Association for Computational Linguistics.

Daumé, III, Hal and Daniel Marcu. 2006. Bayesian query-focused summarization. In *ACL*, pages 305–312, Morristown, NJ, USA. Association for Computational Linguistics.

Elming, Jakob. 2006. Transformation-based corrections of rule-based mt. In *EAMT-2006: 11th Annual Conference of the European Association for Machine Translation*, pages 219–226.

Hermjakob, Ulf, Kevin Knight, and Hal Daumé III. 2008. Name translation in statistical machine translation - learning when to transliterate. In *Proceedings of ACL-08: HLT*, pages 389–397, Columbus, Ohio, June. Association for Computational Linguistics.

Ittycheriah, Abraham and Salim Roukos. 2007. Direct translation model 2. In Sidner, Candace L., Tanja Schultz, Matthew Stone, and ChengXiang Zhai, editors, *HLT-NAACL*, pages 57–64. The Association for Computational Linguistics.

Ji, Heng and Ralph Grishman. 2007. Collaborative entity extraction and translation. In *International Conference on Recent Advances in Natural Language Processing*.

Kirchhoff, Katrin, Owen Rambow, Nizar Habash, and Mona. Diab. 2007. Semi-automatic error analysis for large-scale statistical machine translation systems. In *Proceedings of the Machine Translation Summit IX (MT-Summit IX)*.

Knight, Kevin and Ishwar Chander. 1994. Automated postediting of documents. In *AAAI*, pages 779–784.

Li, Chi-Ho, Dongdong Zhang, Mu Li, Ming Zhou, and Hailei Zhang. 2008. An empirical study in source word deletion for phrase-based statistical machine translation. In *StatMT '08: Proceedings of the Third Workshop on Statistical Machine Translation*, pages 1–8, Morristown, NJ, USA. Association for Computational Linguistics.

Ma, Wei-Yun and Kathleen McKeown. 2009. Where's the verb?: correcting machine translation during question answering. In *ACL-IJCNLP '09: Proceedings of the ACL-IJCNLP 2009 Conference Short Papers*, pages 333–336, Morristown, NJ, USA. Association for Computational Linguistics.

Menezes, Arul and Chris Quirk. 2008. Syntactic models for structural word insertion and deletion. In *EMNLP '08*, pages 735–744, Morristown, NJ, USA. Association for Computational Linguistics.

Papineni, Kishore, Salim Roukos, Todd Ward, and Wei jing Zhu. 2002. Bleu: a method for automatic evaluation of machine translation. In *ACL*, pages 311–318.

Parton, Kristen, Kathleen R. McKeown, James Allan, and Enrique Henestroza. 2008. Simultaneous multilingual search for translingual information retrieval. In *CIKM 08*, pages 719–728, New York, NY, USA. ACM.

Parton, Kristen, Kathleen R. McKeown, Bob Coyne, Mona T. Diab, Ralph Grishman, Dilek Hakkani-Tür, Mary Harper, Heng Ji, Wei Yun Ma, Adam Meyers, Sara Stolbach, Ang Sun, Gokhan Tur, Wei Xu, and Sibel Yaman. 2009. Who, what, when, where, why?: comparing multiple approaches to the cross-lingual 5w task. In *ACL-IJCNLP '09*, pages 423–431, Morristown, NJ, USA. Association for Computational Linguistics.

Snow, Rion, Brendan O'Connor, Daniel Jurafsky, and Andrew Y. Ng. 2008. Cheap and fast—but is it good?: evaluating non-expert annotations for natural language tasks. In *EMNLP '08*, pages 254–263, Morristown, NJ, USA. Association for Computational Linguistics.

Vilar, David, Jia Xu, Luis Fernando D'Haro, and Hermann Ney. 2006. Error analysis of machine translation output. In *International Conference on Language Resources and Evaluation*, pages 697–702, Genoa, Italy, May.

Zhang, Yuqi, Evgeny Matusov, and Hermann Ney. 2009. Are unaligned words important for machine translation ? In *Conference of the European Association for Machine Translation*, pages 226–233, Barcelona, March.