MT Detection in Web-Scraped Parallel Corpora

Spencer Rarrick
University of Washington
Department of Linguistics
PO Box 354340,
Seattle, WA 98195
rarricks@uw.edu

Chris Quirk
Microsoft Research
One Microsoft Way
Redmond, WA 98052
chrisq@microsoft.com

Will Lewis
Microsoft Research
One Microsoft Way
Redmond, WA 98052
wilewis@microsoft.com

Abstract

The Web is an invaluable source of parallel data, but in recent years it has become polluted with increasing amounts of machine-translated content. Using such data to train an MT system can introduce error and decrease the resulting quality of the system. In this paper, we present an algorithm for filtering machine-translated content from Web-scraped parallel corpora, and discuss its application in cleaning such corpora for use in training statistical machine translation systems. We demonstrate that our algorithm is capable of identifying machine-translated content in parallel corpora for a variety of language pairs, and that in some cases it can be very effective in improving the quality of an MT system. Trained on our filtered corpus, our most successful MT system outperformed one trained on the full, unfiltered corpus, thus challenging the conventional wisdom in natural language processing that "more data is better data"[1].

1 Introduction

Extraction of parallel corpora from bilingual websites has proven a valuable means of acquiring training data for use in statistical machine translation (SMT), cross-lingual information retrieval, and various other multi-lingual NLP applications. Several systems have been developed to identify parallel documents on the Web (Nie and Cai, 2001; Resnik and Smith, 2003; Uszkoreit et al., 2010). These systems do well at identifying pairs of documents that are roughly equivalent in structure and information content. However, this kind of content often contains parallel text that is of inferior linguistic quality, most notably content that was generated by a machine translation system. This paper describes a supervised learning approach to improving the utility of Web-extracted corpora by detecting and excluding machine-translated or low-quality document pairs.

The amount of machine-translated content on the Web varies by language. For high-density languages such as English, Japanese, and German, only a small percentage of web pages are generated by machine-translation systems. Among pages for which we identified a parallel document, at least 15% of the sentence pairs annotated for both English-German and English-Japanese appear to contain disfluent or inadequate translations.

Language	% MT	Language	% MT
Lithuanian	51.53%	Slovenian	25.47%
Latvian	50.07%	Hungarian	12.93%
Romanian	47.40%	Estonian	12.13%
Slovak	46.40%		

Table 1: Percentage of the Web that is MT for Various Low-Density Languages

The amount of MT content on the Web rises sharply for lower density languages such as Latvian, Lithuanian and Romanian. Table 1 lists the estimated percentage among *all* Web content (not just bilingual) that is generated by machine translation for various low-density languages. These data were gathered in a previous unpublished study by our team. Latvian and Lithuanian had the highest percentages, with each over 50%. These languages

[1] This quote is generally attributable to (Brants and Xu, 2009). Although they were referring specifically to language models, their comment is also applicable to translation models, particularly those build over large amounts of web-scraped data.

suffer from the scarcest supply of parallel corpora to begin with, so the addition of Web-scraped content has the potential to significantly increase the available amount of data. As the average quality of the data for these languages is relatively low, these are also the languages for which we expect our filter to have the greatest impact.

2 Related Work

Antonova and Misyurev (2011) attempted to detect machine-translated content in a Web-scraped parallel English-Russian corpus using a special phrase-based decoding algorithm designed to find an MT-like reordering of a given reference translation. Sentences with high n-gram similarity to the reordered references have a high probability of being MT. Their algorithm is effective at detecting MT content in English-Russian data, but they note that it may be less effective on language pairs with more similar word order. MT systems trained on the filtered corpus showed only a small improvement in BLEU (Papineni, 2002) over a random baseline.

Other work in MT detection was mostly aimed at finding new methods for automatic MT evaluation. As human translations are considered to be of much higher quality than MT, the task of MT evaluation can be recast as that of determining how "human-like" some MT output is. Several researchers have thus framed MT evaluation as a classification problem, where the quality of a translated sentence is judged to be proportional to the classifier's confidence that it is human-translated.

Corston-Oliver et al. (2001) developed a decision tree classifier designed to determine whether a sentence was human-translated or machine-translated, without need for reference translations. Their model uses two main groups of features: (1) perplexity measures from a lexicalized language model, and (2) various linguistic features, such as branching properties of parses, and the number of pre- and post-modifiers found in the sentence. They evaluated their system using a corpus of 180,000 English sentences (half human-translated from Spanish, and half machine-translated) and were able to significantly outperform the baseline.

Gamon et al. (2005) developed a system that combined scores from an n-gram language model with those output by an SVM classifier to identify "highly disfluent or ill-formed sentences". The specific features extracted from the parses differ somewhat from (Corston-Oliver et al., 2001): the system extracts part of speech tag trigrams, context-free grammar productions, and a number of semantic features such as definiteness of noun phrases, semantic relationship between parent and child nodes, and semantic modification relations. Their system achieved a correlation with human judgment on translation quality that was somewhat lower than BLEU (0.42 to 0.58 for BLEU), but did so without the use of reference translations.

While we use some techniques similar to those discussed in Coston-Oliver et al. and Gamon et al. in this other research, our goal is to identify and filter low-quality content from a large corpus. We are primarily interested in applying the algorithm to parallel data that has been scraped from the Web, as most other parallel data is presumably known to be human-translated.

These factors allow us to exploit a few additional sources of information, but also impose some constraints on what techniques are available. Because our algorithm operates on pairs of webpages, we have access to the URL and full HTML of the target pages, both of which may contain clues to the quality of the translation that are not contained in the text of the documents. We have annotated document pairs in a distribution similar to that of the underlying data, which tells us the correct proportion of positive and negative examples. Furthermore the annotation process tells us something about how pervasive the problem of MT content is on a per-language pair basis.

Additionally, for most parallel webpages, we expect the page to either be entirely human-translated or entirely MT. This allows us to aggregate information at the document level, rather than make decisions at the sentence level. Finally, we are also able to use features that incorporate information from both sides of the document pair.

On the downside, because we intend to apply the filter to a large number of language pairs, we must use language-agnostic techniques where possible. We thus limit ourselves to only those NLP resources that are necessary to build an SMT system or are otherwise language independent: n-gram language models, word breakers, word aligners, and maximum entropy learners/classifiers. We avoid relying on properties of specific languages. Finally, we hope to classify very large corpora (generally at least several hundred thousand docu-

ment pairs per language pair), so we must ensure that our solution is efficient and scalable.

3 System Overview

The core of our algorithm is a maximum entropy classifier. It assigns scores to candidate document pairs based on its confidence that the translation is adequate and fluent on both sides. The term "side" here refers to the half of the document pair that comes from one of the two languages under consideration. Our system is fed these document pair objects by our Web extractor, which is inspired by STRAND (Resnik and Smith, 2003), but which has significant improvements.

Each document pair object consists of the following data:

- URL of each side of the web page
- Full HTML for each side
- A list of aligned sentence pairs
- Sentence-broken text for each side
- *Static Rank* for each side [2]

Some features used by the document-level classifier are derived from a second, sentence-level maximum entropy classifier, which scores all sentence-pairs found in each document pairs by a sentence aligner. The interaction between the two classifiers is depicted in Figure 1.

3.1 Features

Features used by the sentence and document level classifiers are divided into several groups. In the feature ablation experiments described in §5.1, each of these feature groups is included or excluded from the feature set as a unit. Unless otherwise specified, each feature template is applied to each side of the sentence or document pair.

Sentence Level Features:

- *General*
  - character and token counts, ratio between sides
  - mean token length, ratio between sides
  - sentence length bucket indicator features[3]
- *Out-of-Vocabulary (OOV)*
  - total number of OOV tokens per side

[2] *Static Rank* is a measure of relative importance of a web page, used by Bing in search indexing.

[3] 16 possible combinations: 1, 2, 3-6, or >6 tokens per side

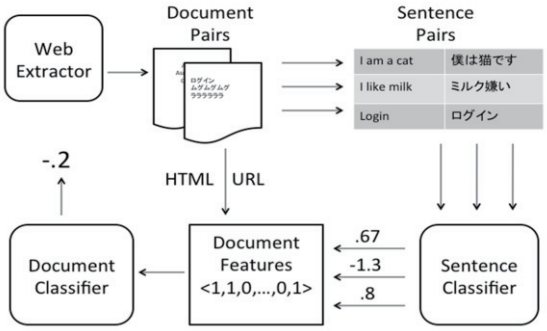

Figure 1 - System Flowchart

  - count of OOV tokens containing only or some "alphabetic characters" [4]
  - tokens seen on both sides but OOV for one
- *Lexical*
  - binary indicator features for unigrams
- *Script*
  - binary indicator features for each script type (e.g. "Latin", "Cyrillic", "Hiragana")
  - count, ratio of characters of each script (before and after discounting *Common*[5] script)
  - binary indicator feature for ellipsis
- *Token Match*
  - count and ratio for each token type[6] that does not have an exact match on other side
  - lexicalized features for unmatched tokens
  - indicator features if all or no tokens of a type have exact matches

In addition to these five groups, we also experimented with additional feature groups that used word alignments, n-gram language model perplexities, function words, and suffixes. However, inclusion of these features did not improve performance over the feature groups listed, and they are not discussed here in depth.

Document Level Features:

- *Basic*
  - number of aligned sentence pairs
  - total number of sentences on each side
  - ratio of sentences that have an alignment
  - ratio of number sentences between sides

[4] "Alphabetic characters" are defined by Unicode regular expressions and are not limited to the roman alphabet.

[5] *Common* characters include whitespace, certain punctuation marks found across languages, and numerals.

[6] Token types: words, punctuation, and numerals

- *Static Rank* and ratio between sides
- binary indicator feature for presence of translation marker in HTML[7]

- *URL*
  - protocol type (e.g. "http:" or "https:")
  - binary indicator features for URL domain
  - binary indicator features for each token appearing in the domain or entire URL
  - character, token count for domain, URL
  - count of each punctuation type in URL
- *Sentence Score*
  - mean and sum of scores for aligned sentence pairs, weighted three ways: uniformly, by token count and by character count
  - count, ratio of sentences in each score range or "bucket".

In the *Sentence Score* feature group, we use fourteen sentence score buckets: $x \leq -1.0$, $x > 3.0$, and 12 uniformly sized buckets from -1.0 to 3.0. These values were tuned by hand.

4 Data

We evaluate our system on four language pairs: English-Latvian, English-Romanian, English-Japanese and English-German. Japanese and German are high-resource languages, while Latvian and Romanian are relatively low-resource languages, for which a substantial portion of all Web content was determined to be machine translated.

For each of these language pairs, we annotated 200 randomly sampled document pairs from among those identified by our Web extractor as parallel document candidates. Annotators were first asked to make an initial assessment as to whether each pair of documents appeared to be parallel. They could answer "YES", "NO", or "YES-BUT-BAD" in the case that the documents shared structure and some content but one was clearly machine-translated or otherwise disfluent.

For document pairs marked "YES" or "YES-BUT-BAD", they were then asked to annotate several aligned sentence pairs that had been randomly sampled from the document pair. For each sentence pair, they assessed the fluency of each side and the adequacy of the translation (i.e. whether the meaning was preserved).

[7] Such as the Google Translate URL

Sentence pairs were treated as "human-translated" if they were marked "YES" for fluency on both sides as well as adequacy. Generally, we treated document pairs as "human-translated" if 80% of the sampled sentences were human-translated, with each sentence weighted by the number of tokens. For English-Romanian, however, a very large percentage of document pairs were annotated as "YES-BUT-BAD" at the document level, and so we gave preference to the annotator's judgment here over our heuristic.

We used duplicate annotations to evaluate inter-annotator agreement. For all language pairs, we found Kappa scores above 0.6, which Landis and Koch (1977) consider to constitute "substantial agreement." Given the difficult and rather subjective nature of the annotation task, we feel that scores in the observed range are strong.

5 Experiments

Our experiments evaluate two aspects of our algorithm's performance. The first is its ability to distinguish machine-translated content and other low quality translations from clean human translations. The second is the impact of filtering a corpus on the quality of the statistical machine translation system trained on that data.

5.1 Performance on the Detection Task

To evaluate our algorithm's ability to distinguish MT from human-translated content at the sentence and document levels, we employed five-fold cross-validation on the human-annotated data set, as well as human evaluation of our classifier's predictions on unannotated data. For document-level tests, we first set aside half of the annotated documents and used their aligned sentence pairs to train a sentence-level classifier for use in extracting the *Sentence Score* features. For cross-validation tests, we evaluated both overall accuracy and 11-point average precision. 11-point average precision is better at capturing the quality of the classification throughout the range of confidence scores.

Tables 2, 3, and 4 report accuracy and 11-point average precision for our cross-validation experiments for the English-Japanese, English-Latvian, and English-German annotated data sets. We have not run this set of experiments for our English-Romanian data. The "Baseline" row is simply the percentage of positive instances in each test set.

The first six rows after "Baseline" show the effect of varying the sentence-level features. For these rows, the document-level scores reflect all document-level features groups included (i.e. *Sentence Score*, *Basic*, and *URL*). The last six rows show the effect of varying document-level features. For document-level feature sets that include *Sentence Score*, the sentence-level classifier uses the feature set that had the strongest document-level results with all document-level features turned on (*Lexical* for English-Japanese, and English-German, and *General* for English-Latvian).

For the three language pairs shown here, we see that the classifier significantly outperforms the baseline in all four metrics for at least some feature sets. For sentence-level performance, the *Lexical* feature group alone is at or very close to the maximum score for 11-point average precision, though for English-Japanese and English-Latvian, we see a further increase in accuracy of around 2% by adding the remaining sentence-level features. However, it is surprising that performance of a sentence-level feature set on sentences does not necessarily correlate with performance on the document-level. For English-Latvian, using only the *General* sentence-level feature group led to the best document level performance. For English-Japanese and English-German, the best performing document-level feature sets did not even include the *Sentence Score* feature group.

We performed human evaluation to confirm that our classifier is able to preferentially assign higher scores to human-translated sentence pairs than to machine-translated. For each of our four language pairs, we used our classifier to rank millions of sentences pairs, and sampled 200 sentence pairs at roughly equal intervals in the ranking (the sampled sentences' exact indices in the ranking were randomized within a range). We then randomized the order, and presented them to a human annotator for evaluation. The annotator gave a simple YES/NO judgment for each sentence pair depending on whether they felt it constituted a

Language Pair	Avg Prec	Baseline	Error Reduc
Japanese-Eng	0.94	0.83	65.7%
Latvian-Eng	0.70	0.44	46.9%
Romanian-Eng	0.70	0.57	31.0%
German-Eng	0.87	0.71	55.2%

Table 5: Human Evaluation of Classifier Ranking

Features	Sent AvgP	Sent Acc	Doc AvgP	Doc Acc
Baseline	**.599**	**.599**	**.460**	**.460**
Lexical	**.910**	.829	.640	.640
General	.800	.680	**.800**	**.740**
OOV	.46	.599	.730	.64
Script	.770	.633	.710	.640
Token	.760	.720	.700	.640
All	**.910**	**.846**	.690	.660
Sent Only			.740	.680
URL			.760	.660
Basic			.640	.660
Sent+URL			.790	**.740**
URL+Basic			.740	.640
Sent+Basic			.730	.680

Table 2: English-Latvian Cross-Validation

Features	Sent AvgP	Sent Acc	Doc AvgP	Doc Acc
Baseline	**.833**	**.833**	**.706**	**.706**
Lexical	**.930**	.863	**.900**	.804
General	.900	.828	**.900**	.804
OOV	.700	.833	**.900**	.804
Script	.890	.831	**.900**	.804
Token	.900	.845	**.900**	.804
All	**.930**	**.868**	**.900**	.804
Sent Only			.710	.686
URL			.830	.667
Basic			**.900**	**.863**
Sent+URL			.840	.686
URL+Basic			**.900**	.804
Sent+Basic			.890	**.863**

Table 3: English-German Cross-Validation

Features	Sent AvgP	Sent Acc	Doc AvgP	Doc Acc
Baseline	**.828**	**.828**	**.640**	**.640**
Lexical	**.962**	.877	**.872**	.800
General	.907	.838	.804	.780
OOV	.738	.828	.830	.640
Script	.932	.853	.871	.800
Token	.923	.851	.848	.800
All	.960	**.900**	.770	.740
Sent Only			.850	.720
URL			.750	.700
Basic			.750	.620
Sent+URL			.849	.740
URL+Basic			**.880**	.780
Sent +			.860	.760

Table 4: English-Japanese Cross-Validation

good translation. We then calculated 11-point average precision for each set using the annotations as a gold standard. The results are presented in Table 5. The English-Japanese ranking was generated using the *URL* and *Basic* feature groups and all features were used for the other language pairs.

The baseline column is the expected average precision for a random ordering of the sampled sentences. The results range from a 31.0% error reduction for English-Romanian to a 65.7% error reduction for English-Japanese, confirming that the classifier is in fact quite successful at separating human-translated content from machine-translated.

5.2 Effect of Filtering on Translation Quality

Our final method of evaluation is to compare SMT systems trained on data filtered with our algorithm to a baseline system. For these experiments, we use our classifier to score and rank a very large set of document pairs (hundreds of thousands per language pair, which contain millions of aligned sentence pairs). For each data point, we then select the N sentences with the highest scores and either add them to a trusted, human-translated training set to train an MT system, or train an MT system on them in isolation. Our models were trained using a treelet translation system (Quirk et al, 2005). Finally, we compute BLEU scores for the resulting systems on a variety of test sets.

The MT systems trained on baseline and filtered data sets were evaluated using either one or two test sets. We evaluated BLEU on a newswire test set for each language pair, and for English-Romanian and English-Latvian, we also evaluated on a second, domain-balanced test set. These results are reported in Table 6.

We have several baselines systems for these experiments. The first is a system trained on just the core data set alone. Second, we have systems trained on the same number of randomly sampled sentences from the dataset. Finally, we can compare system quality against a system trained using a large sample of unfiltered, Web-scraped data.

We will use the following nomenclature when describing groups of sentence pairs and the SMT systems trained on those sentence pairs:

- *best:* highest ranked sentences (by classifier)
- *rand:* randomly sample of Web-scraped sentences
- *base:* trusted data not scraped from the Web
- *all:* the set of all Web-scraped sentences

	0	500k	1M	1.5M	2M	All
English-Latvian Balanced Test Set (2.7M)						
BT	21.5	27.3	30.1	29.7	29.9	29.7
RT	21.5	24.3	25.4	27.1	28.4	29.7
B		20.6	23.9	23.8	23.9	24.6
R		16.5	18.9	21.9	23.0	24.6
English-Latvian Newswire Test Set						
BT		13.3	14.4	14.4	15.1	15.2
RT		15.0	14.6	15.1	15.5	15.2
English-Romanian Balanced Test Set (2.45M)						
BT	17.1	27.6	34.0	37.3	40.3	42.1
RT	17.1	28.6	32.2	35.6	38.7	42.1
English-Romanian Newswire Test Set						
BT	19.2	21.3	21.7	22.8	22.8	22.9
RT	19.2	21.9	22.5	23.0	22.7	22.9

	0	1M	2M	3M	4M	All
English-Japanese Newswire Test Set (6.3M)						
BT	12.3	12.3	12.1	12.21	12.7	13.3
RT	12.3	12.8	13.1	13.0	13.2	13.3
B		8.4	9.5	10.6	11.3	13.1
R		11.4	12.1	12.4	12.4	13.1

	0	1M	2M	All
English-German Newswire Test Set (8M)				
BT	12.31	13.01	13.28	15.62
RT	12.31	13.13	13.52	15.62

BT: base + best **RT:** base + rand
B: best **R:** rand

Table 6: Effect or Filtering on BLEU

For example, *"best 1M only"* would refer to a system trained on only the one million highest ranked sentence pairs from the Web-scraped data. *"base + rand 500k"* would refer to a system trained on a non-Web core data set with 500,000 additional sentence pairs that were randomly sampled from Web-scraped data.

The strongest result that we can hope for is to beat the BLEU score of an *all* system using some subset of *best* sentences, as this would show conclusively that we are able to filter out some sen-

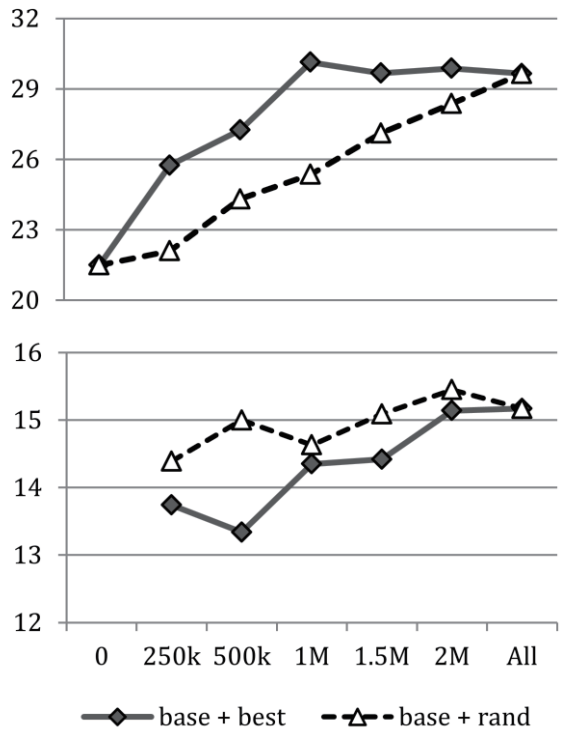

Figure 2 -- Effect of Filtering on BLEU for English-Latvian on Balanced Test Set (above) and Newswire Test Set (below)

tences that are actually harmful to performance. Beating a *rand N* system with a *best N* system by a large margin would be weaker, but still positive result as it would indicate that sentences ranked highly by our classifier are on average more useful as training data than a random sample.

Note that success on the detection task does not guarantee success on this metric. The primary motivation of this research was the hypothesis that the presence of machine-translated content in our training data was having a net negative impact on our MT systems. However, it is quite possible that machine-translated sentence pairs could contain useful vocabulary items not seen elsewhere in the training corpus. It is also possible that a human-translated sentence pair might come from a domain unrelated to the test set, or contain only words that are already frequently seen elsewhere, thus being relatively unhelpful as training data.

We saw our strongest result with the English-Latvian system on the balanced test set (See Figure 2). The *base + best 1M* system outperformed *base + rand 1M* by nearly 5 BLEU points and surpassed the *base + all* system (an additional 1.7 million sentence pairs) by 0.4 BLEU points. The English-Romanian system evaluated on the balanced test set also showed somewhat positive results. The *base + 1M* system outperformed *rand + 1M* by 1.8 BLEU points. However, none of the *base + best* systems was able to beat *base + all*.

Despite strong performance on the balanced test sets, *best* and *base + best* systems for all language pairs were outperformed by corresponding *rand* systems on newswire test sets. We have identified two factors that may have led to this discrepancy. First, it appears that our filter introduces a domain bias into the corpus. Many of our features are correlated with domain, leading our classifier to select documents from domains with a high proportion of human-translated documents. For example, presence of the word *"argument"* indicates that a page is likely to be tech domain, and therefore professionally localized rather than MT. We believe that the domain bias introduced here causes *best* systems to perform poorly on news domain.

The second factor is vocabulary diversity. As our classifier assigns scores at the document level, all aligned sentence pairs from a document pair will have the same score and appear together in the ranking. This effectively reduces the number of documents that *best* sentence pairs are drawn from. Furthermore, the classifier tends to assign similar scores to similar document pairs. The *best* data sets therefore tend to include sentence pairs with redundant vocabulary items at the expense of those with novel vocabulary. *Rand* sentence pairs, on the other hand, are sampled uniformly from the pool of all sentence pairs in the corpus. Therefore, despite the fact that they contain some MT, these sentences have a more even distribution across documents and domains, and therefore better vocabulary coverage than the analogous *best* set.

We have attempted some variations on these end-to-end tests in hopes of improving the benefit of our filter across domains and language pairs. We suspect that some of our *best* systems may be performing poorly because of vocabulary in the low-ranked sentence pairs that has been thrown away. Accordingly, these variations attempt to minimize the loss in vocabulary, while still mitigating the impact of disfluent sentences.

Our first variation was to keep low ranked sentences that contain rare vocabulary items. We do so by iterating through sentences pairs in order of classifier score, and tallying count for each lexical item encountered. If a token has been seen less

frequently than some threshold, the entire sentence is included in the training set, and otherwise it is removed. Low-scoring (likely MT) sentences will be visited last, and kept only if they contain tokens seen infrequently in higher scoring sentences. The English-Romanian system trained using this method outperformed the full system on the newswire test set by 0.54 BLEU points. However, we saw no improvement for the other systems.

The other variation was to build a second mapping table with the low-scoring sentences, in addition to the one generated from our trusted data and *best* sentences from the Web-scraped data, similar to the approach used by Axelrod et al. (2011) for domain adaptation. Weights for the two tables were tuned on a development data set. Intuitively, the "low-quality" table would be given a low weight and only affect the output when no appropriate phrase is found in the main table. Using this method, we saw a boost for English-Latvian of 0.59 BLEU points on the newswire test set over the full system. However, this same system was much weaker on the balanced test set, so it appears that there may once again be some domain effects.

6 Conclusions

We have developed an algorithm that is able to identify machine-translated content in a Web-scraped parallel corpus using only a small amount of human-annotated training data. In some cases, MT systems trained on our filtered corpora were extremely strong (mostly notably English-Latvian on the balanced test set). We feel that this result is quite significant, as it shows that it is possible to improve performance of an MT system by *removing* large amounts of training data.

In other cases, however, using these filtered corpora has failed to improve the quality of the resulting MT system. As confirmed by the small gains seen in (Antonova and Misyurev, 2011), using MT detection to improve BLEU is not always straightforward. Our next step is to find ways of making our algorithm more consistently beneficial across domains and language pairs. Thus far, we have explored a few alternative ways to apply the filter and preliminary results are promising.

In addition to machine translation, MT detection also has potential application in search engine indexing. It may be desirable to rank machine-translated pages below human-written ones. While some adaptation would be necessary to apply the classifier to monolingual documents rather than parallel documents, we believe that our general approach is applicable.

References

Alexandra Antonova and Alexey Misyurev. 2011. *Building a Web-Based Parallel Corpus. Proceedings of ACL.*

Amittai Axelrod, Xiaodong He and Jianfeng Gao. 2011. *Domain Adaptation via Pseudo In-Domain Data Selection. Proceedings of EMNLP.*

Thorsten Brants, Peng Xu. 2009. *Distributed Language Models. Proceedings of HLT-NAACL (Tutorial Abstracts).*

Simon Corston-Oliver, Michael Gamon and Chris Brocket. 2001. *A Machine Learning Approach to the Automatic Evaluation of Machine Translation. Proceedings of ACL.*

Michael Gamon, Anthony Aue and Martine Smets. 2005. *Sentence-level MT evaluation without reference translations: Beyond Language Modeling. Proceedings of EAMT.*

J. Richard Landis and Gary G. Koch. 1977. *The Measurement of Observer Agreement for Categorical Data. Biometrics*, Vol. 33, No. 1

Juan-Yun Nie and Jian Cai. 2001. *Filtering noisy parallel corpora of Web pages. Proceedings of IEEE.*

Kishore A. Papineni, Salim Roukos, Todd Ward and Wei-Jing Zhu. 2002. *BLEU: A Method for Automatic Evaluation of Machine Translation. Proceedings of ACL.*

Chris Quirk, Arul Menezes and Colin Cherry. 2005. *Dependency Treelet Translation: Syntactically Informed Phrasal SMT. Proceedings of ACL.*

Philip Resnik and Noah A. Smith. 2003. *The Web as a Parallel Corpus. Computational Linguistics*, Vol. 29, 349-380.

Jakob Uszkoreit, Jay Ponte, Ashok Popat and Moshe Dubiner. 2010. *Large Scale Parallel Document Mining for Machine Translation. Proceedings of COLING.*