Predicting MT Quality as a Function of the Source Language

David M. Rojas[1], Takako Aikawa[2]

[1]Butler Hill Group and Indiana University
[2]Microsoft Research
One Microsoft Way
Redmond, WA 98052, USA
{v-drojas, takakoa}@microsoft.com

Abstract

This paper describes one phase of a large-scale machine translation (MT) quality assurance project. We explore a novel approach to discriminating MT-unsuitable source sentences by predicting the expected quality of the output.[1] The resources required include a set of source/MT sentence pairs, human judgments on the output, a source parser, and an MT system. We extract a number of syntactic, semantic, and lexical features from the source sentences only and train a classifier that we call the "Syntactic, Semantic, and Lexical Model" (SSLM) (cf. Gamon et al., 2005; Liu & Gildea, 2005; Rajman & Hartley, 2001). Despite the simplicity of the approach, SSLM scores correlate with human judgments and can help determine whether sentences are suitable or unsuitable for translation by our MT system. SSLM also provides information about which source features impact MT quality, connecting this work with the field of controlled language (CL) (cf. Reuther, 2003; Nyberg & Mitamura, 1996). With a focus on the *input* side of MT, SSLM differs greatly from evaluation approaches such as BLEU (Papineni et al., 2002), NIST (Doddington, 2002) and METEOR (Banerjee & Lavie, 2005) in that these other systems compare MT *output* with reference sentences for evaluation and do not provide feedback regarding potentially problematic source material. Our method bridges the research areas of CL and MT evaluation by addressing the importance of providing "MT-suitable" English input to enhance output quality.

1. Introduction

Various metrics for automatic MT evaluation have been proposed (BLEU, METEOR, etc.). The focus of evaluation, however, has been on the target side of MT and has been used primarily during system development since the usually required reference sentences are not available at runtime in a production setting. We argue that the impact of the source on the quality of the target should be addressed, and that CL might play a critical role not only for content providers but also for developers debugging an MT system

The motivations for our current approach are several. First, as MT system developers, we have seen that the quality of an output varies depending on a number of characteristics of the input sentence. For instance, the longer an input sentence is, the lower the quality of its translation. Further, we know that the more complex the structures of an input are, the worse the output will be. Our aim is to develop a model that can reflect such intuitions. Second, if the predicted difficulty of translating the source gives us an indication of the amount of post-editing needed on the target MT sentence, a score from SSLM can function as an estimate for this post-editing. This will help increase localizer productivity and decrease localization costs. For instance, we might stratify the compensation method for localization depending on the complexity of documents and the expected difficulty of post-editing, as opposed to simply using a flat per-word pricing method. Third, we would like to equip content providers with appropriate linguistic feedback during the authoring or editing phase so that they can make changes that will improve MT quality. Content providers typically are not aware of the types of sentence structures that yield lower quality (machine) translations. If we provide them certain guidelines in advance or feedback while they write, the overall quality of MT output should improve.

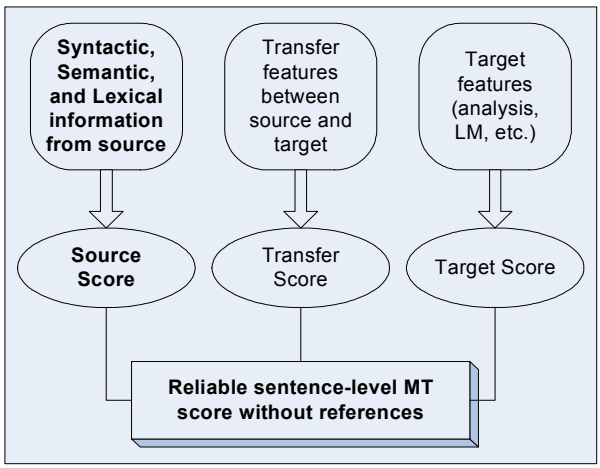

Figure 1. MT Evaluation Framework with Emphasis on Source Score Component

Although our present focus, via SSLM, is on assessing source quality pre-MT, it is only one module in our MT quality assurance project. We have also been working on models that consider features of different components of our MT system and exploring a way to combine SSLM with these other models, so that we can optimize the predictability of the quality of our MT system. Figure 1 provides a high-level overview of the evaluation project architecture and the relative contribution of SSLM.

[1] By "unsuitable" we are simply referring to source sentences that are predicted to produce low quality translations. We do not assume *a priori* that source "(un)suitability" equates with "(un)grammaticality."

In the remainder of the paper, we first present a brief overview of our NLP and MT systems in Section 2. Then we describe types of features extracted from our parser for training SSLM. Section 3 furnishes details on our training data and the design of our experiments. In Section 4 we give the results of our experiments, and in Sections 5 we provide some observations on selected features from SSLM. In Section 6 we present our future research directions, along with our concluding remarks.

2. System Overview and Feature Extraction

2.1. Overview of our NLP and MT Systems

Our MT system ("MSR-MT") consists of four major components: (i) Analysis (Parser); (ii) Logical Form Creation (Heidorn, 2000); (iii) Alignment/Transfer (Menezes & Richardson, 2001); and (iv) Surface String Generation (Aikawa et al., 2001; Corson-Oliver et al., 2002). For details on MSR-MT, see Richardson (2004).

Here we focus on the first two modules, as they are closely related to SSLM features. Based on the syntactic analysis provided by our parse, we create what we call "Logical Form" (LF), which is a basic semantic representation of a given input. Figure 2 and Figure 3 provide, respectively, the analysis of our parser and the corresponding LF for the sentence "The files or folders cannot be copied."

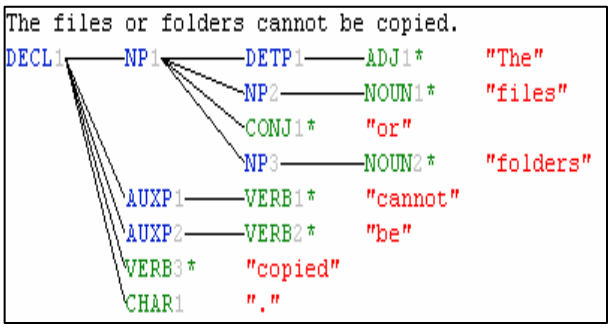

Figure 2. Sample Parse from NLP Engine

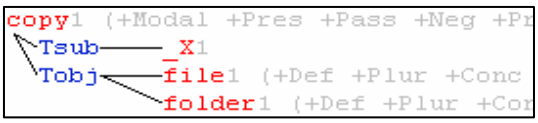

Figure 3. Sample Logical Form from NLP Engine

2.2. SSLM Feature Extraction

From the parse and LF described above we extracted a number of count features from source (English) sentences. The following list a sample of SSLM features.

- *Syntactic Features*: NumParses (possible parse analyses) for a given sentence, subordinate clauses, complement clauses, conjoined NPs, conjoined VPs, modals, be-verbs, commas, colons, MWEs (multi-word expressions), Fitted (failure of the parse analysis), NumTokens (the number of tokens in a given sentence), etc.
- *Semantic/LF Features*: passive, progressive, ECM, comparatives, superlatives, etc.
- *Lexical Features*: expert-identified problematic lexical items such as morphologically negative adjectives (e.g., improbable, unnecessary, unwilling) and negative adverbs (e.g., hardly, seldom), etc.

The total number of features used for our experiment is 146.[2] We contend that such features are likely to contribute to the ambiguity or complexity of the source and thus impact the translation process, for better or worse. Feature engineering is ongoing.

3. SSLM: Training Data and Experiments

3.1. Training Data

We used sentences drawn from Microsoft technical manuals and documents to train and test SSLM for the following five language pairs:

English→Japanese	(EJ)	2422 sentences
English→Spanish	(ES)	2498 sentences
English→Italian	(EI)	1395 sentences
English→German	(EG)	1433 sentences
English→French	(EF)	1409 sentences

The training data were first translated into these languages via MSR-MT and then rated by humans with respect to the acceptability of their translations using the following scale, with no other instructions concerning how they should make their judgments:

4 : Ideal: Not necessarily perfect, but grammatically correct; all information accurately transferred.

3 : Acceptable: Not perfect (stylistically or grammatically odd), but definitely comprehensible, AND with accurate transfer of all important information.

2 : Possibly Acceptable: Possibly comprehensible (given enough context and/or time to work it out); some information transferred accurately.

1 : Unacceptable: Absolutely not comprehensible and/or little or no information transferred accurately.

We then extracted features for each sentence as described in Section 2.2.

3.2. Experiments

Our basic assumption is that the human evaluation score on MSR-MT output can serve as a proxy for our ability to translate the input. Therefore, for each language pair, we model the "translatability" of a sentence, indicated by the human evaluation judgment, as a weighted linear combination of the sentential features in a simple linear regression model:

$$\text{HumanScore} = \beta_0 + \beta_1 * \text{NumParses} + \beta_2 * \text{subordinate clauses} + \beta_3 * \text{complement clauses} + \ldots + \beta_{146} * \text{negative adverbs}$$

To evaluate the general performance of such a model and the degree to which the model score suggests how well our system translates the input, we created twenty random training/test samples from the data for each language, estimated a model for each training set, obtained model scores for each sentence in each test set, and calculated the Pearson correlation between the SSLM

[2] We provide a full list of features used in the poster session.

predictions and the human scores. Given the novelty of our approach in considering only source information, we compare SSLM results with those from an English language model (LM), a model that also only considers source information. The English LM was trained on 1.2M in-domain sentences (not overlapping with the SSLM data). Perplexity scores were obtained for only one subset of test data, since we did not expect there to be a strong correlation between natively written English sentences and scores from a language model trained on data from the same domain. It is important to note that we cannot compare SSLM performance with BLEU, METEOR, or other evaluation metrics. These others compare the test translation with one or more references while SSLM never has access to reference sentences, given the context in which it is applied.

4. Results

Figure 4 compares correlation results for human scores with both SSLM and the source language model. While still not very strong, SSLM correlation results are consistently better than those of the language model, suggesting that linguistic characteristics about the source do contribute information regarding the eventual quality of the translation, and that the relationship is stronger than that of simple n-gram frequencies.

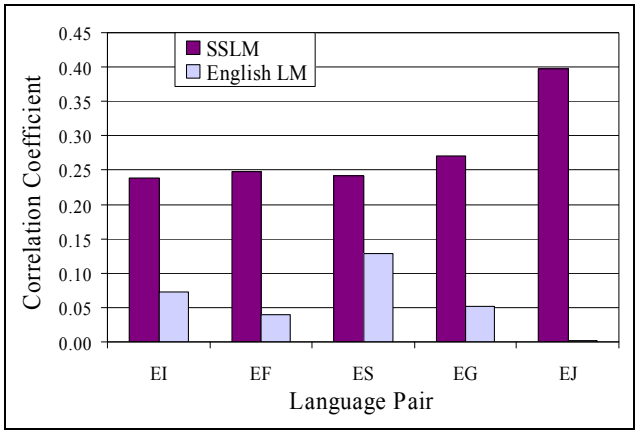

Figure 4. Correlation Coefficients for Human Evaluation Scores with LM & SSLM

Recasting SSLM as a classifier whose role is to determine whether sentences are suitable for translation by our system or not, we obtain the performance depicted in Table 1. Since there is no point of comparison for such figures, we state them here as a baseline. One factor to bear in mind when evaluating a model such as SSLM is that editors and authors are likely to have a low tolerance for false-positives. A consideration like this stresses the importance of working together with content providers to determine their needs and the possible applications of the model.

	EJ	ES	EI	EF	EG
Precision	0.653	0.830	0.641	0.582	0.665
Recall	0.836	0.937	0.719	0.784	0.905
F_β	0.733	0.880	0.676	0.667	0.766

Table 1. SSLM Classification Performance

5. Observations on the Selected Features

As mentioned at the outset of the paper, one motivation for SSLM is to construct a model that reflects developers' intuitions and offers linguistic feedback to content providers regarding the suitability/translatability of source sentences. To this end, we present in Table 2 sample lists of features selected as significant during modeling for the five language pairs examined.

EJ	ES
NumTokens	**NumTokens**
NumParses	**NumParses**
Comma adjunct count	MWE
Passive	Comma adjunct count
BE verb count	**Fitted**
Progressive	
ECM	
Fitted	

EI	EF	EG
NumParses	**NumParses**	**NumTokens**
MWE	Comma adjunct count	**NumParses**
Colon	Conjoined NP count	MWE
BE verb count	WH lemma count	Passive
Fitted	ECM	**Fitted**
	Fitted	Cleft

Table 2. Sample of Features Selected as Significant

We now make some observations regarding some of the features, focusing here only on three that were selected in at least 16 of the 20 random trials for most language pairs: (i) NumTokens, (ii) NumParses, and (iii) Fitted.[3] The feature *NumTokens* refers to the number of tokens in a sentence. The fact that this feature is consistently selected is intuitively reasonable: the longer a given input sentence is, the lower the quality of our MT output for that sentence. The feature *NumParses* refers to the number of possible analyses based on our parser. The value of NumParses describes on a certain level the degree of ambiguity associated with a given sentence: the greater the number of possible parses available for a given sentence, the more ambiguous the sentence is. For instance, the sentence "The system failed to delete the file on the Web" is two-way ambiguous with respect to PP (prepositional phrase) attachment: the PP "on the Web" could be attached (i) to the NP "the file" or (ii) to the VP "failed to delete". The value of NumParses for this sentence is therefore 2. On the other hand, if an author uses the sentence, "On the Web, the system failed to delete the file," the sentence contains no ambiguity; and hence the value of the NumParses for this sentence is 1. The feature *Fitted* is binary and is true if our parser fails to find a spanning parse for the sentence. The fact that this feature affects output quality is rather obvious, given the parser-dependent translation architecture described in 2.1: if the parser cannot analyze the sentence, the output will not be translated properly.

[3] In the poster session we discuss in more detail why certain features may have been selected for a particular language pair.

6. Future Directions and Conclusion

As depicted in Figure 1, SSLM is merely one component of our ongoing MT quality assurance project, and we plan to tackle a number of issues with respect to SSLM in the upcoming months. First we would like to add more linguistic features—in particular, those further characterizing the ambiguity of the source. Second, we would like to more carefully investigate language-specific features as well as cross-linguistic features. As previously mentioned, one project goal is to supply content providers with appropriate linguistic guidance/guidelines so that MT output quality can be improved. We are working with language experts to determine why certain features are consistently selected for particular language pairs. Third, we would like to see whether SSLM can predict the suitability of the source for different types of MT systems (e.g., RBMT/SMT), not just for the MT system described here.

In addition, we will continue development on the Target Score component (cf. Figure 1), incorporating insight from Gamon et al. (2005), Liu & Gildea (2005), and Rajman & Hartley (2001), all of whom use various features from the target to rate or rank the output of a system. We are currently working on an approach similar to work by Uchimoto et al. (2005), who use information from reverse translations of the MT output to make decisions about sentential elements that are not translatable. After obtaining reasonable results for the individual model scores, we can explore ways to combine all the models described in Figure 1 in order to optimize the confidence we have in MT quality from our system. The MT quality assurance architecture demonstrates the autonomy of the models: since each considers a different stage of the MT process (i.e., source, alignment, and target), the models are to some extent independent of each other and can accordingly be used either separately or collectively, depending on the needs of end users.

We have shown that SSLM, albeit a very simple model, can help determine whether a sentence should be submitted to our system for translation or not. Furthermore, SSLM provides insightful information about some of the features that influence MT quality. This characteristic is important in that it spans the content provider, localization, and MT communities. We believe that SSLM, even in its incipient stage, demonstrates the potential contribution of controlled language in enhancing MT quality and stresses the importance of effective quality control as a pre-cursor to MT.

7. References

Martin, L.E. (1990). Knowledge Extraction. In *Proceedings of the Twelfth Annual Conference of the Cognitive Science Society*. Hillsdale, NJ: Lawrence Erlbaum Associates, pp. 252-262.

Aikawa, T, M. Melero, L. Schwartz, and A. Wu. (2001). Sentence generation for multilingual machine translation. In *Proceedings of the MT Summit VIII*, Santiago de Compostela, Spain.

Banerjee, Satanjeev and Alon Lavie. (2005). METEOR: An automatic metric for MT evaluation with improved correlation with human judgments. *ACL'05 Workshop on Intrinsic and Extrinsic Evaluation Measures for Machine Translation and/or Summarization*.

Blatz, John, Eric Fitzgerald, George Foster, Simona Gandrabur, Cyril Goutte, Alex Kulesza, Alberto Sanchis, Nicola Ueffing. (2004). Confidence estimation for machine translation. In *Proceedings of COLING 2004*, pp. 315-321.

Corston-Oliver, Simon, Michael Gamon, Eric Ringger, and Robert C. Moore. (2002). An overview of Amalgam: A machine-learned generation module. In *Proceedings of the International Natural Language Generation (INLG) Conference*. New York, USA. pp. 33-40.

Gamon, Michael, Anthony Aue and Martine Smets. (2005). Sentence-Level MT evaluation without reference translations: beyond language modeling In *Proceedings of EAMT 2005*.

Heidorn, G. (2000). Intelligent writing assistance. In R.Dale, H.Moisl and H.Somers (eds.), *A Handbook of Natural Language Processing: Techniques and Applications for the Processing of Language as Text*. New York: Marcel Dekker, pp. 181-207.

Liu, Ding and Daniel Gildea. (2005). Syntactic features for evaluation of machine translation. *ACL'05 Workshop on Intrinsic and Extrinsic Evaluation Measures for Machine Translation and/or Summarization*.

Menezes, Arul and Stephen D. Richardson. (2001). A best-first alignment algorithm for automatic extraction of transfer mappings from bilingual corpora. In *Proceedings of the Workshop on Data-driven Machine Translation at 39th Annual Meeting of the Association for Computational Linguistics*, Toulouse, France, pp. 39-46.

Nyberg, E. H. and T. Mitamura. (1996). Controlled language and knowledge-based machine translation: Principles and practice, In *Proceedings of the First International Workshop on Controlled Language Applications*.

Papineni, Kishore A., Salim Roukos, Todd Ward and Wei-Jing Zhu. (2002). BLEU: A method for automatic evaluation of machine translation. In *Proceedings of ACL 2002*, pp. 311-318.

Quirk, Christopher. (2004). Training a sentence-level machine translation confidence measure. In *Proceedings of LREC 2004*, pp 825-828.

Rajman, Martin and Tony Hartley. (2001). Automatically predicting MT systems rankings compatible with Fluency, Adequacy or Informativeness scores. In *Proceedings of the MT Summit VIII*, Santiago de Compostela, Spain.

Reuther, U. (2003). Two in one – Can it work? Readability and translatability by means of controlled language. *Controlled Language Application Workshop (CLAW-03)*, pp. 124-132.

Richardson, Steve (2004). Machine translation of online product support articles using a data-driven MT system. In *Proceedings of AMTA 2004*, pp. 246-251.

Uchimoto, K., N. Hayashida, T. Ishida, H. Isahara. (2005). Automatic rating of machine translatability. In *Proceedings of MT Summit X*, pp. 235-242.