Context in Machine Translation

Aaron B. Phillips
Advanced Machine Translation Seminar

Language is not static. The words we select to convey a concept *and* the semantics of individual words change based on subject, audience, and one's cumulative life experience. The wonder of it all is that language is still an effective and meaningful mode of communication. With regard to machine translation, this is a troubling scenario. Computers excel at deterministic tasks; modeling a phenomena as fluid and capricious as language is challenging.

Language is disambiguated through additional information commonly called 'context'. Given a particular statement, context is a vague and malleable notion describing things outside of that statement that can elucidate it. WordNet (Fellbaum, 1998) defines context as "discourse that surrounds a language unit and helps to determine its interpretation" and, "the set of facts or circumstances that surround a situation or event". In this paper I consider context to be any information beyond the sequence of lexical words that form a statement to be translated. While humans may refer to a logical argument or prior knowledge as context, these are difficult concepts for a machine to work with. As such, this paper will focus on the use of linguistic features to identify context. These include, but are not limited to: part of speech (POS), lemmas, chunk markers, nearby words, sentence information, document information, and genre.

Clearly context alone is not sufficient for translation. Thus, the goal of this work is to understand how context can be *added* as an *additional* source of information to *existing* translation models.

1 Machine Translation, From Ages Past to Ages Present

Machine translation has been developed in a variety of paradigms, but recently data-driven techniques have dominated the field. The data-driven approach, especially that represented by statistical machine translation (SMT) will be the presumed translation paradigm for this paper. SMT originally followed the "noisy channel model". This generative story describes the translation process as recovering English text that has been passed through a noisy channel, thereby distorting it. Following Bayes Rule, it can be succinctly represented as:

$$P(e|f) = P(e)P(f|e)$$

This equation neatly splits the translation process into two components, one that models the probability of the target sentence, $P(e)$, commonly referred to as the language model, and a translation model that computes the probability of a source word given a target word, $P(f|e)$.

Early SMT systems were word-oriented. They treated each word of the input as independent translation units. The only part of the system that captured context was the language model. By selecting a fluent sequence of translations, the language model enforced (to some degree) the selection of a coherent (contextual)

meaning across the sentence as a whole. However, the effect of this depends heavily on the strength of the language model. A trigram language model, for example, will only preserve context across a small window of words. Additionally, this only addresses consistency within the target sentence and does not model context present in the source.

The field made a leap forward when researchers began using phrases (of which words are a proper subset) as basic translation units. These phrasal translations are often decomposable and theoretically the same target sequence can be modelled by a word-based system. However, the key advantage to using phrases is that the translation model can provide a probability for a group of words, in context, rather than separate probabilities for each word, in abstract. A careful reader will argue that this is not contextual information as context was defined earlier as "information beyond the sequence of lexical words". However, this definition is influenced by modern phrase-based models where the phrase (or sequence of lexical words) is the basic translation unit. Compared to a word-based model, including phrases in the translation process incorporates the context of the neighboring words when forming a translation unit. As my definition suggests, this type of context will be presumed by the work covered later in the paper. All the work reviewed in this paper utilizes phrase-based systems, so I will investigate techniques that model context beyond the basic phrase. However, it is important to acknowledge that jointly modeling a sequence of words is a much stronger predictor than modeling each word independently and that this proto-context is already present in phrase-based systems.

A more recent advancement in MT has been the move away from the "noisy channel model" to a more general log-linear model. This discriminative approach to MT allows for an arbitrary number of features to be combined together. Each translation is scored with the equation

$$s(t) = \prod_i \phi_i(t)^{\lambda_i}$$

where $\phi_i(t)$ is a feature of the translation t and λ_i is its weight. A n-gram language model is integrated into this framework by adding a language model feature with dependencies on neighboring target words. Similarly, one approach to integrating context that we will see later is to include context features with dependencies on neighboring source words.

Local nuances and alterations in meaning that occur within a monotonic unit are usually captured well by modern phrase-based MT systems. However, as with word-based systems before them, there is not always sufficient evidence present within the translation unit to generate an unambiguous translation. What these systems currently lack is the ability to address context external to the phrase (translation unit). Modeling longer and longer phrases can overcome this limitation, but this approach quickly becomes impractical. The translation model can only effectively learn translations of phrases that occur frequently in the training corpus. Due to a finite set of resources, unseen or infrequent phrases are unlikely to be translated correctly by phrase-based systems. The hope is that context surrounding the phrase can be used to aid the translation process; translations for phrases will be dynamically adjusted based on their context, eliminating or reducing the need to match extremely long lexical sequences. This will allow for use of shorter (thus more frequent) translation units resulting in a more robust model.

2 Integrating Context

In machine translation we are not only concerned with deciphering *what* is being said, but also *how* that concept should be restated in another language. This roughly reflects the division between the translation model and the language model present in most modern MT

systems. (Even though modern systems have moved away from the "noisy channel model", MT systems are still typically thought of as consisting of these two separate components.)

The translation model describes the "what"–it identifies target phrases that correlate well with each source phrase. This model can be thought of as selecting the most likely meaning of each source phrase, where the collection of target phrases defines the set of possible meanings. This model is trained from a bilingual corpus where information about neighboring phrases in both the source and target are available. To this end, there is a lot of additional context information available to the translation model, that modern phrase-based systems ignore.

The language model describes the "how"–it is responsible for identifying a sequence of target phrases such that the result is fluent and coherent. Most language models are n-gram based and calculate the probability of a sequence based only on the lexical words. Context can be addressed within these models by modeling the probability of additional information (such as POS or lemma tags) or increasing the size of the history. It is important to note, however, that this is explicitly a monolingual process. The language model does not depend on the source text and cannot model any contextual information about the source text.

While context is important in both of these models, the focus of this paper will be on integrating context within the translation model. This is not to belittle the importance of the language model; changes to the language model might even show more dramatic improvement than those to the translation model. However, to narrow the scope of this paper I must select only one and I see a greater potential for novelty and integration of context in the translation model because both the source and target are available for use. I will investigate what contextual features are appropriate to use, how to score them, and how these new features should be integrated in the translation model.

2.1 Context as Word Sense Disambiguation

Word sense disambiguation (WSD) attempts to select the proper 'sense' of a word given a particular sentence. This area of research focuses explicitly on how to model the meaning of a word given its context. A commonly cited article giving further background is (Yarowsky and Florian, 2002). WSD is a traditional machine learning classification problem. Features are selected that might guide the selection of a word sense such as lemmas, POS, sentence bag-of-words, neighboring words, etc. A simple approach is to train a Bayesian classifier from a large amount of data to predict the sense of a word given the surrounding features. Using the same features, sophisticated statistical models can be built as well. Indeed, (Carpuat et al., 2004) employs an ensemble of naive Bayes, maximum entropy, boosting, and kernel PCA for the classification task.

WSD is an attractive paradigm for integrating context within the translation model. Not only have effective features and models already been identified for WSD, but the goals of WSD and the translation model are very similar. The crucial difference is that WSD seeks to identify the proper sense while the translation model attempts to identify the proper translation of a word. However, as (Vickrey et al., 2005) points out, if each possible target translation is treated as a unique sense, then this process is the same. The additional discrepancy is that WSD is usually done at the word level, so some changes may have to be made to predict phrases in context.

A series of papers that have all taken this approach are grouped together below. By building upon current research in WSD, they attempt to add new features to the translation model that inform the model about the particular context. In contrast to a traditional SMT system, these models are no longer static. Each occurrence of

a phrase pair has a unique context and must be handled separately.

2.1.1 An Unimpressive Beginning

One of the first published attempts at integrating a stronger model of context within the translation model, (Carpuat and Wu, 2005) was a failure. Later I will discuss more successful approaches, but this work is insightful because it indicates that not all approaches work equally well. (Carpuat and Wu, 2005) demonstrates that a SMT system results in significantly lower performance than the state of the art WSD system *at a WSD task*. From this they conclude that the ability of the SMT system to model context is weak and they hope to see improvement by integrating the WSD system within the SMT system.

Their WSD system was trained using Senseval-3 Chinese lexical data which covers 20 words. 175 sentences were extracted from NIST MT04 for evaluation that contained at least one of the known ambiguous Chinese words. The researchers tried two different approaches in order to test the effectiveness of integrating a WSD system with a SMT system. The first approach used the WSD system to separately model word selection and then constrain the SMT phrase table to only permit entries in agreement with the WSD system. Translations were only considered valid if they matched a possible English gloss of the sense of the Chinese word predicted by the WSD system. The second approach translated complete sentences and then selectively replaced ambiguous target words using a gloss of the sense predicted by the WSD system. The first method performs better than the second, but as shown in Table 1 both fail to outperform the baseline SMT system.

The authors observe that one problem could be that the WSD lexicon is significantly smaller than the vocabulary of the SMT system. The result is that when the WSD is used in conjunction with the SMT system, the possible hypotheses are too constrained. To overcome this issue, they retrain the WSD using the larger vocabulary in the SMT lexicon. Unfortunately, this lowered the BLEU score even further from 0.1239 to 0.1232.

MT System	BLEU
SMT Baseline	0.1310
SMT + WSD Filtering	0.1239
SMT + WSD Post-Processing	0.1253

Table 1: Results from (Carpuat and Wu, 2005)

The authors also introduce a problem referenced later by other researchers known as the "language model effect". Changing the selection of one word in a sentence can significantly alter the surrounding words due to different language model probabilities. Even if the system using WSD selects a better translation, this translation could harm the overall sentence if the language model can less reliably predict an appropriate sequence of words using the better translation. This was the motivation behind the second approach to perform post-processing, thereby preventing the language model from altering the sentence. Even though some of the lexical choices were superior, a review of the translations showed that for all but two of the target words, the system using WSD failed to increase the BLEU score. BLEU is very sensitive to the "language model effect" because it especially favors long matches with a reference. While human evaluation observed this to be an issue, even calculating BLEU at the unigram level resulted in a lower score for the system using WSD. Thus, the system using WSD, on average, did not even select more words that were contained in the references.

This work has been criticized for using a state-of-the-art WSD system, (Carpuat et al., 2004), but a translation system was not state-of-the-art at the time the paper was published (ISI's ReWrite decoder (Germann, 2003)). While this may have contributed to their negative results, more fundamentally the problem with this work

was that they poorly integrated the WSD system with the SMT system. The WSD system made hard decisions that only removed possible translations from the SMT system. The translation model and language model work together to find a translation that is both meaningful and fluent. Hard filtering of potential candidates from the translation model (unless it can be done near perfectly) disrupts this balance and limits the ability of the language model to build a fluent translation. This work demonstrates that a deeper integration is necessary–one that alters the weight of translations in a probabilistic manner–and does not simply chop off potential candidates.

2.1.2 Changing the Paradigm

(Vickrey et al., 2005) changed the paradigm of how to use a WSD system with regard to machine translation. (Carpuat and Wu, 2005) used WSD in the traditional sense that classified each Chinese source word to a unique sense. For the translation task, each Chinese sense was mapped to an English target word using HowNet glosses. Recognizing that the goal is translation, not sense identification, (Vickrey et al., 2005) modeled the problem more directly. They specified that each target word represents a different sense of the source word. Not only does this approach avoid the lossy lexicon-lookup, but it also conflates senses that are represented by the same target word (where ambiguity can be preserved).

The pseudo-WSD/context model built by (Vickrey et al., 2005) is a rather simple logistic regression model that uses POS and surrounding lexical words as features. This model was able to predict the translation of a single word with an accuracy of 0.605. Selecting the most frequent translation yielded an accuracy of 0.526. Both models were trained on the French-English Europarl and evaluated on a held-out subset containing 1859 ambiguous words.

Instead of integrating this within a real MT system, the researchers chose a more synthetic approach. Using an aligned corpus, they removed a single word in the target sentence and attempted to predict it using their classifier. In addition they included probabilities from a language model for the resulting target sentence. This combination is akin to the translation model and language model combination present in most MT systems. The baseline system utilized the language model prediction log-linearly combined with frequency-based estimates of $P(s|t)$ and $P(t|s)$ learned from the alignment links. Using 655 sentences from the Europarl with 3018 missing target words, the baseline system correctly predicted the target word with an accuracy of 0.833. Including the context model as an additional feature boosts the accuracy to 0.846. While the improvement is very small, the authors argued that these are under-estimates because multiple possible translations are usually valid while the scoring only considered the single word present in the reference translation as correct (only one reference translation was available).

(Giménez and Màrquez, 2007) extends the work of (Vickrey et al., 2005) to handle *phrasal translations* and moves from the synthetic blank-filling task to using a real MT system. Like (Vickrey et al., 2005), this work builds a pseudo-WSD system with each target representing a distinct class, but models phrases, not just words.

To train a context-informed model (Giménez and Màrquez, 2007) uses the full corpus as examples of when a particular phrase pair is expected and when it is not. From the corpus they extract a large number of features detailing sentence and document context associated with each phrase pair. With regard to local context they extract 1-gram, 2-gram, and 3-grams of words, POS, lemmas, and base chunk labels within a five word window of each phrase pair. To model global context, topical information is collected by treating each source sentence as a bag of lemmas. These features along with positive and negative examples of phrase pairs are

fed into a set of local linear SVMs that perform one-vs-all classification. Last, softmax is applied to the SVM score to produce a translation probability. While this is very similar to WSD, the researchers call this Discriminative Phrase Translation (DPT) as the focus is on translation, not sense disambiguation.

(Giménez and Màrquez, 2007) trained their models using the English-Spanish Europarl corpus and evaluated on a held-out set of 1008 sentences. Unfortunately, due the number of features and possible contexts, DPT was only accurate with phrase pairs that occurred very frequently. In order to address this deficiency, they back-off to using MLE in order to estimate phrase pairs occurring less than 50,000 times. Thus, DPT was only used to predict 41 phrases, covering 25.5% of their test corpus. Automatic metrics as shown in Table 2 give high scores to the translations, but DPT was only slightly better than or equal to the baseline. (They also developed their own metric to justify their results, but even there the improvement was minimal.)

MT System	BLEU	METEOR
$P(e) + P_{MLE}(f\|e)$	0.59	0.77
$P(e) + P_{MLE}(e\|f)$	0.62	0.77
$P(e) + P_{DPT}(e\|f)$	0.62	0.78

Table 2: Automatic Evaluation from (Vickrey et al., 2005)

A more informative follow-up human evaluation reviewed 114 sentences where the DPT and MLE predictions differed and at least 5 words in the sentence had a DPT prediction. Table 3 shows the counts of how many times the MLE or DPT sentence was selected as being better in terms of adequacy, fluency, and overall. These results indicate that DPT improves adequacy but lowers fluency. Averaging both scores suggested that humans slightly preferred the DPT translations.

These results only used monotone decoding, which artificially diminishes the quality of the MT system. Additionally, the baseline translation model only used $P(f|e)$ or $P(e|f)$. This is a considerably weak baseline, and they could have at least log-linearly combined both scores. This definitely raises questions as to the quality of the results and whether they could be reproduced with a state-of-the-art translation system. These qualifications aside, (Giménez and Màrquez, 2007) is the first work to describe a complete MT system that uses context to probabilistically guide the translation model.

	Adequacy	Fluency	Overall
MLE $>$ DPT	39	**84**	83
MLE $=$ DPT	100	76	46
DPT $<$ DPT	**89**	68	**99**

Table 3: Human Evaluation from (Vickrey et al., 2005)

2.1.3 Tight Integration

A slightly different approach was taken by (Chan et al., 2007) using Hiero. Unlike a traditional phrase-based SMT system, Hiero uses a very shallow grammar derived from the phrase pairs. Decoding is performed by applying these grammar rules bottom-up to generate the input sentence. After each grammar rule is applied, (Chan et al., 2007) appends two context features to the feature set of the current span. This is done in a dynamic fashion during decoding when a lexical sequence is formed because some of Hiero's rules contain non-terminals. The first feature used represents the contextual probability of the translation given the source words of the span under consideration based on an external WSD classifier. The WSD classifier in this work was only trained on unigrams and bigrams. Therefore, in order to obtain the contextual probability of longer phrases, they combined the contextual probabilities of shorter phrases using a heuristic algorithm that greedily selects the combination resulting in the highest WSD probability. Additionally, not all words or phrases will be represented by the WSD classi-

fier. The second feature is inversely exponential to the length of the translation chosen by the WSD system, $exp(-|t|)$. With a negative weight, this feature will reward rules that use translations suggested by the WSD (and thereby offset the fact that some phrases are not represented by the WSD).

The WSD system was built using a SVM with local collocations, POS, and surrounding words as features. They followed the approach of (Lee and Ng, 2002) and report that their system is comparable to the best performing system in Senseval-3, (Carpuat et al., 2004). Hiero with and without the context features was trained on the Chinese-English FBIS corpus. Evaluation on NIST MT03 results in a baseline of 29.73 BLEU and 30.30 BLEU when using the context features. The reader should note that the baseline Hiero system is *much* stronger than the MT systems used by (Giménez and Màrquez, 2007) and (Carpuat and Wu, 2005), making the improvement all that more impressive. This is the first work that tightly integrates context into the translation model such that it is just like every other feature and can be optimized through minimum error rate tuning (MERT).

Following up on their initial negative results, a few years later, (Carpuat and Wu, 2007) also provides a successful and tight integration of context within the translation model. Like (Chan et al., 2007), they introduce new context features that can be tuned just like any other feature within the log-linear translation model. Instead of using Hiero and incorporating these dynamically during decoding, (Carpuat and Wu, 2007) uses a standard phrase-based SMT system and appends the context features to the phrase table. A side-effect of this approach is that the phrase table must now represent the particular context of each phrase pair and can become quite large. Learning from the work that followed their first paper, the authors now disambiguate between phrases and not just words as in the Senseval tasks. Additionally, they use target phrases from the corpus to represent the sense of each source phrase. Unlike previous work, they do not actually build or use an external WSD system to generate a single feature or probability distribution. Instead, they plug into the SMT log-linear model their collection of WSD features. The individual features they report using are bag-of-words context, local collocations, position-sensitive local POS, and basic dependency features.

A highlight of this work is that they perform a very comprehensive evaluation. Their system yielded only modest gains, but improvement was consistent with evaluation performed using eight different metrics and four different test sets. The largest experiment, trained on 1 million sentences from Chinese-English newswire and evaluated on NIST MT04, is shown in Table 4. These results are not directly comparable with (Chan et al., 2007) since (Carpuat and Wu, 2007) does not specify from what corpus the training data was extracted and uses NIST MT04 instead of NIST MT03 for evaluation. Nonetheless, it is interesting that (Chan et al., 2007) has scores 10 BLEU points higher than (Carpuat and Wu, 2007).

(Carpuat and Wu, 2007) also experiments with artificially limiting their model to only add context features to single word entries in the phrase table. This did not result in reliable improvement in translation quality and they suggest this was the downfall of their first work, (Carpuat and Wu, 2005).

2.1.4 Summary

From this progression of work one trend is clear: a naive integration of a WSD system and a MT system is not very beneficial. Rather, features from the WSD need to be tightly integrated as features of the translation model so that they can be optimized in conjunction with the rest of the translation model features during MERT. To this end, the last two works, (Chan et al., 2007) and (Carpuat and Wu, 2007), are most promising, although the works by (Vickrey et al., 2005) and (Giménez and Màrquez, 2007)

MT System	BLEU	NIST	METEOR	METEOR (no syn)	TER	WER	PER	CDER
Baseline SMT	20.20	7.198	59.45	56.05	75.59	87.61	60.86	72.06
SMT + WSD	20.62	7.538	59.99	56.38	72.53	85.09	58.62	68.54

Table 4: Evaluation from (Carpuat and Wu, 2007)

prepared the way for them. The main difference between (Chan et al., 2007) and (Carpuat and Wu, 2007) is that they are using different decoders, and therefore, the stage (phrase table or on-the-fly during decoding) at which they integrate context changes slightly. (Carpuat and Wu, 2007) also eliminates the external WSD system by incorporating the WSD features directly into the translation system. While conceptually this is cleaner and seems to be more appropriate, it is actually (Chan et al., 2007) who reports the best performing system (and best improvement from using context).

While there has been a clear progression and solution for how to integrate context, there is still a large degree of variance among researchers regarding which features to use to identify the context. The context features are all borrowed from WSD and some like POS are used in all systems. However, the particular set of features keeps changing from work to work. One thing lacking in all the works reviewed that would have been beneficial is an empirical study of how each context feature affects MT. (Carpuat and Wu, 2007) comes the closest to doing this, and reports that the POS immediately outside the phrase and bag-of-words full sentence context are the most important features. However, they conclude this based on which features have the largest log-linear weights from MERT. While this is interesting, it is not sufficient justification as the weights only indicate how much a feature must be scaled and do not directly indicate its importance. In the same way that different MT systems frequently use different features for translation (and some features work better for particular systems), perhaps there is no "correct" set of context features and this is something that must be tailored to each particular MT system.

2.2 Context as Information Retrieval

The second approach to modeling context is more implicit. Instead of altering the translation model to identify specific context features for each phrase pair, one alters the data from which the translation model is built so that it better reflects the context of the text to translate. This technique, more commonly known as adaptation, is effective at globally skewing the translation model toward a particular target. Data-driven statistical models are merely a reflection of the text they are trained on. Filtering or re-weighting the training data will alter the model's predictions. One can produce better contextual translations by emphasizing training data that reflects the same nuances (read context) as the text they plan to translate. This is usually most effective at a high level such as genre. For example, when translating newswire text, as a last resort one might use a translation from an internet blog, but it is preferable to select a translation from another newswire source.

2.2.1 Filtering

(Hildebrand et al., 2005) applies information retrieval (IR) techniques to filter a training corpus such that it is maximally similar to the text to be translated. For every input sentence to be translated the n-most-similar sentences are extracted from the corpus. The n-most-similar sentences for all input sentences are combined together (potentially including duplicates) to form a new training corpus. A standard translation model can then be built from this filtered training corpus. In order to calculate sentence similarity, the authors measure the cosine distance between TF-IDF term vectors–a common

algorithm for document similarity in IR.

It might seem odd that after identifying similar sentences the authors lump everything together and build a single, static translation model. The authors justify this decision by explaining that at the high-level, the characteristics of a document such as genre are unlikely to change quickly. Although building a unique translation model for each input sentence or group of sentences might model context better, such a focused translation model is likely to be brittle and not yield robust probabilities.

(Hildebrand et al., 2005) evaluated this approach on 506 lines of Chinese-English tourism dialogue. 20,000 lines of tourism dialogue and 9.1 million lines of newswire and speeches were available for training. Table 5 shows the results of their experiments under several different scenarios. The scores for the n-most-similar sentences are the optimal values on the test set after evaluating the system at n = 10, 20, 30, 40, 60, 70, 80, 100, 125, 150, 175, 200, 250, 300. The authors also experimented with using perplexity to automatically select the appropriate value for the n, but it was unclear whether their technique was successful.

2.2.2 Weight Adjustment

Trying to exploit the same idea, (Brown, 2005) presents an online, dynamic approach within the EBMT framework. The general idea is that when examples are retrieved from the corpus that match the input document, one should give an additional bonus to the examples that also share the same context as the input. The author breaks this down into two categories: local context and inter-sentential context. In order to exploit local context, (Brown, 2005) recognizes that when an example is largely the same as the input sentence then there will be numerous examples (1-gram, 2-gram, 3-gram, etc.) retrieved from the same training sentence. Therefore, for each input sentence, one gives a bonus to examples that are retrieved from the same training sentence.

Language	Local	Inter-Sent.	Both
French	**+1.51%**	+0.33%	-0.26%
Chinese	**+0.83%**	-0.33%	+1.08%
Spanish	+1.22%	-0.60%	-0.28%

Table 6: Evaluation from (Brown, 2005)

Second, inter-sentential context models consistency among input sentences. This is handled by recording usage counts of which examples are used to translate each input sentence. When an example is selected that has been used frequently, or it is *near* an example that has been used frequently for translation, then the example is given a bonus. These two strategies help weight examples with contextual similarities to the input document more heavily than the rest of the examples in the corpus. This work uses the same basic idea present in (Hildebrand et al., 2005), but implements it in a dynamic fashion that can alter each sentence separately. Unfortunately, evaluation did not show this technique to yield strong improvement. Table 6 displays the relative improvement when using each type of context matching. Some language pairs did perform slightly better with the contextual bonuses, but only slightly, and only those marked in bold were statistically significant.

2.2.3 Mixture Model

Instead of building one adapted translation model, (Foster and Kuhn, 2007) builds several translation models and weights each model based on its similarity to the input. The authors take the training corpus and split it into several smaller corpora. In their experiments they used labeled genre information to split the corpus, but they suggested this could also be done through automatic topic identification and clustering. Separate context-specific translation models are built from each smaller corpus using standard methods. A global translation model is built from a weighted combination of the individual context-specific translation models. If

MT System	BLEU	NIST
20k in-domain	0.4621	8.1129
20k in-domain + 15k random out-of-domain	0.4850	8.2262
20k in-domain + 75k random out-of-domain	0.4501	7.9482
Optimal n-most similar	0.4871	8.2132

Table 5: Evaluation from (Hildebrand et al., 2005)

the development set is very similar to the test set, then the weights applied to each translation model can be learned directly through tuning. A more interesting case, however, is to abstract this so that the weights of each context-specific translation model are a function of a similarity metric. With this method, the mixture weights dynamically change for each input document. (Foster and Kuhn, 2007) identified four features to use to measure similarity: TF-IDF, LSA, perplexity, and the total probability of the mixture model. Their results showed that the adaptation works in the static context (when the development set is similar to the test set), but no significant improvement was shown when using the dynamic context matching. This suggests that the similarity algorithm was not adequate at automatically weighting the translation models.

2.2.4 Summary

All of these approaches work by favoring data that is most similar to the input. (Hildebrand et al., 2005) creates a hard cut-off based on sentence similarity while (Brown, 2005) takes a softer approach that takes advantage of the EBMT framework to merely boost the weight of examples drawn from similar documents. The difficulty with such an approach within SMT is that the phrase table is built from a training corpus and remains static during translation. (Foster and Kuhn, 2007) tries to overcome this limitation by building several static phrase tables and interpolating dynamically between them during decoding.

The common thread among all this work is the attempt to alter the translation model based on IR similarity metrics. This globally skews the translation model and implicitly addresses some of the same context issues that were explicitly modeled in "context as word sense disambiguation". The advantage to this approach (with the exception of the EBMT system in (Brown, 2005)) is that it is mostly external to the translation model and does not require complex code adjustments to implement. The general idea followed by these researchers is laudable and has potential. Unfortunately, all the work in this area struggled to show improvement. This is partially to be expected because "context as information retrieval" models the problem more indirectly than "context as word sense disambiguation", which only resulted in modest improvement.

3 To the Future and Beyond

From this progression of research, where can we project the integration of context within the translation model is heading?

As mentioned earlier, log-linear models provide an excellent framework in which disparate pieces of information from many different sources can be combined together. Both (Chan et al., 2007) and (Carpuat and Wu, 2007) incorporate context as features within log-linear models. I expect this to be the most common approach in the future because of the prevalence of MT systems using log-linear models and the relative ease with which context can be added to them. The good news, then, is that modeling context will not require a new paradigm for MT.

In order to perform experiments that incorporate context, researchers performed preprocessing, statically built multiple translation

models, or built extremely large translation models representing all possible forms of context. While effective for experimentation, these techniques are more or less "hacks". The reason for this is most modern phrase-based SMT systems are not designed to model dynamic information. To simplify the model, features are only dependent on the translation unit. This allows the statistical model to be pre-computed prior to translation. Building a static model is simpler and faster, but its' predictive capabilities are also more limited. Hopefully, as computational power increases, the simpler static models will be less necessary. Further advances in machine translation that allow models to be built online, dynamically during translation (that use features outside the translation unit) will be significantly advantageous for researchers attempting to incorporate context. Yet, there is a notable chicken-and-the-egg problems here: it is hard to justify building dynamic, online translation systems without (yet) seeing a significant benefit brought about by including dynamic context features.

To this end, when dynamic, online systems become available, the two approaches outlined above, "context as word sense disambiguation" and "context as information retrieval" will likely converge. Similar to the work of (Brown, 2005), IR-oriented features could be generated dynamically. Then IR-oriented and WSD-oriented techniques could work alongside each other to identify the proper context. Indeed, (Carpuat and Wu, 2007) and (Giménez and Màrquez, 2007) report extracting a feature based on treating the sentence as a bag-of-words which bears remarkable similarity to some of the IR-oriented features (although neither paper specifies how the bag-of-words were used). Additionally (Brown, 2005)'s calculation of local context is very similar to the local collocations feature present in WSD-oriented systems of (Chan et al., 2007) and (Carpuat and Wu, 2007). While systems built within the "context as word sense disambiguation" paradigm show better result at the present, the two approaches should complement each other well, as WSD is more focused on phrase-level context and IR-oriented features are more focused on document-level context.

One consistent, albeit depressing fact (with the possible exception of (Chan et al., 2007)) is that a lot of effort has been expended with very minimal to modest improvements according to automated evaluation metrics. As a result, many researchers blame the evaluation metrics. BLEU in particular is criticized heavily for its over-reliance on long n-grams that occur in the reference. While there *are* significant problems with automated evaluation metrics, the metrics themselves not entirely at fault. (Giménez and Màrquez, 2007) and (Carpuat and Wu, 2005) include human evaluations and (Carpuat and Wu, 2007) uses a battery of eight different automated metrics. These more extensive evaluations did actually correlate with the findings of BLEU. The general trend was that context improved adequacy, but decreased fluency of the output. This situation–improving one aspect of the translation while worsening another–is especially difficult to judge. Indeed, the particular combination of adequacy and fluency desired may depend on the audience or task. It is not until these systems are capable of improving both adequacy *and* fluency that using context will be clearly beneficial.

In conclusion, there is a colloquial expression in Computer Science that describes this situation: "Garbage In, Garbage Out". Our systems and underlying algorithms are only as good as the data on which they are trained. Including contextual information only provides a *little* more information to the models than the sequence of lexical words. Furthermore, many phrases are unambiguous or occur infrequently enough that the context cannot be accurately modeled. While context is indeed useful, one should not expect it to radically transform the translation model. Context is certainly not garbage, but our expectations for context

may be too high. If context is only predictive of subtle nuances, then one should only expect subtle nuances of the output to change. These types of changes will not–and should not be expected–to dramatically alter evaluation metrics (either automated or human). That being said, such subtle nuances will be necessary for machine translation to be pervasive and adopted for real tasks. Therefore research in context must continue, because its not a question of *whether* context is useful, but *when* will context be useful. Human evaluations consistently showed that adequacy–expressing the intended meaning–improved. Subtle nuances that context does alter may not be evident or may not be useful until machine translation passes some quality threshold. If a sentence is generally incoherent, then correctly translating a subtle nuance of an embedded phrase will go unnoticed. Thus, it is possible that machine translation is not quite at a high enough quality level for contextual information to be a significant factor.

References

Ralf Brown. 2005. Context-sensitive retrieval for example-based translation. In *Proceedings of Machine Translation Summit X*, September.

Marine Carpuat and Dekai Wu. 2005. Word sense disambiguation vs. statistical machine translation. In *Proceedings of the 43rd Annual Meeting on Association for Computational Linguistics*, pages 387–394. Association for Computational Linguistics, June.

Marine Carpuat and Dekai Wu. 2007. Context-dependent phrasal translation lexicons for statistical machine translation. In Bente Maegaard, editor, *Proceedings of Machine Translation Summit XI*, September.

Marine Carpuat, Weifeng Su, and Dekai Wu. 2004. Augmenting ensemble classification for word sense disambiguation with a kernel pca model. In *Proceedings of Senseval-3, Third International Workshop on Evaluating Word Sense Disambiguation Systems*. SIGLEX, Association for Computational Linguistics, July.

Yee Seng Chan, Hwee Tou Ng, and David Chiang. 2007. Word sense disambiguation improves statistical machine translation. In *Proceedings of the 45th Annual Meeting of the Association for Computational Linguistics (ACL 2007)*. Association for Computational Linguistics.

Christiane Fellbaum, editor. 1998. *WordNet: An Electronic Lexical Database (Language, Speech, and Communication)*. The MIT Press, May.

George Foster and Roland Kuhn. 2007. Mixture-model adaptation for smt. In *Proceedings of the Second Workshop on Statistical Machine Translation*, pages 128–135. Association for Computational Linguistics, June.

Ulrich Germann. 2003. Greeedy decoding for statistical machine translation in almost linear time. In *HLT-NAACL 2003*.

Jesús Giménez and Lluís Màrquez. 2007. Context-aware discriminative phrase selection for statistical machine translation. In *Proceedings of the Second Workshop on Statistical Machine Translation*, pages 159–166. Association for Computational Linguistics, June.

Almut Silja Hildebrand, Matthias Eck, Stephan Vogel, and Alex Waibel. 2005. Adaptation of the translation model for statistical machine translation based on information retrieval. In *Proceedings of the European Association of Machine Translation (EAMT 2005)*, May.

Yoong Keok Lee and Hwee Tou Ng. 2002. An empirical evaluation of knowledge sources and learning algorithms for word sense disambiguation. In *Proceedings of the ACL-02 conference on Empirical methods in natural language processing*, pages 41–48. Association for Computational Linguistics.

David Vickrey, Luke Biewald, Marc Teyssier, and Daphne Koller. 2005. Word-sense disambiguation for machine translation. In *Proceedings of the conference on Human Language Technology and Empirical Methods in Natural Language Processing (HLT-EMNLP 2005)*, pages 771–778. Association for Computational Linguistics.

David Yarowsky and Radu Florian. 2002. Evaluating sense disambiguation across diverse pa-

rameter spaces. *Natural Language Engineering*, 8(4):293–310.