ABSTRACT

Word based translation models in particular and phrase based translation models in general assume that a word in any context is equivalent to the same word in any other context. Yet, this is not always true. The words in a sentence are not generated independently. The usage of each word is strongly affected by its immediate neighboring words. The state-of-the-art machine translation (MT) methods use words and phrases as basic modeling units. This paper introduces Context Dependent Words (CDWs) as the new basic translation units. The context classes are defined using Part-of-Speech (POS) tags. Experimental results using CDW based language models demonstrate encouraging improvements in the translation quality for the translation of dialectal Arabic to English. Analysis of the results reveals that improvements are mainly in fluency.

1. INTRODUCTION

Word based translation models rely on the assumption of pairwise correspondence between the words of the target and source sentences. This is done by considering all word pairs for a given sentence pair. Word pair dependency models are known as alignment models. The phrase based alignment models [2] built on the word based translation models by using phrases as the basic unit of translation. Phrases were effective in improving the local word reordering and translation of short idioms.

The alignment models are generated based on strong independence assumptions. That is, each target language word generates zero or more source language words and the generated source words are independent of the other source words generated by other target language words. This implies that the context dependencies between source language words in a sentence are not modeled by the alignment models. The syntactic and semantic relationships between words are modeled only in the target language to some extent via the target language model. If we knew the correct semantic meaning of each word in the source language, we could more accurately determine the appropriate words in the target language. While phrase-based systems do take into account context within phrases, they are not able to use context across phrase boundaries. This is especially important when ambiguous words do not occur as part of a phrase, which is composed of a sequence of words that perhaps possess no syntax or semantic meanings, - verbs in particular often appear alone [18].

We propose Context Dependent Words (CDWs) as the new translation modeling units. These units are formed by taking into account the context in which each word appears. The context can be defined via lexical, semantic or syntactic means, such as POS tags. Recalling the evolution of acoustic modeling units in speech recognition will be helpful to illustrate the need for CDW units for MT. In the early days of speech recognition context independent phones were used as the basic modeling units. However, as large quantities of data become available the expressive power of context independent phones limited the potential performance gains because of large amounts of data. In order to improve the modeling capability for speech data context dependent phones were introduced as the basic modeling units [4]. To date they have been used as the basic building blocks in the state-of-the-art speech recognition systems.

We believe CDWs that generate different versions of the same lexical form based on the prior or context knowledge is highly needed. They will help to disambiguate multiple senses of a word within the translation process. Furthermore, using these units may reduce the morphological gap between language pairs when used on the less inflected side and, thus improve the performance since morphological gap is known to degrade the translation performance [6, 17].

The context classes for CDWs can be defined in a number of ways. In this study we chose the POS tags to define them. The reason for using POS tags is simply because of having off-the-shelf POS tagger for English. Syntax based information [11, 9] in general and POS tags in particular have been extensively used to improve MT performance with limited success [6, 7, 8]. The way POS tags are used in this work is entirely different as compared to previous studies.

The rest of the paper is organized as follows. Section 2 describes the CDW modeling for MT. A brief description of the statistical MT architecture is provided in Section 3. The experimental results and discussion are presented in Section 4, followed by the summary of our
findings in Section 5.

2. CONTEXT DEPENDENT WORD MODELING

Many words in any natural language have multiple senses, for example, in English the word “cold” has two main senses and may refer to a disease or a temperature sensation. The specific sense intended is determined by the context in which “cold” appears. In “I am taking medicine for my cold”, the disease sense is intended, while in “It is getting cold here”, the temperature sensation sense is meant. However, in an MT task they are likely to correspond to different words in the target language. This introduces ambiguity to be sorted out by the translation models and eventually may degrade the translation quality.

Using CDW units can mitigate, if not totally eliminate, “word sense ambiguity” effect on the translation performance. Good CDW units should be consistent and trainable. Consistency refers to having similar characteristics for different instances of the same unit. Consistency is important because it improves the discrimination between different CDW units, which governs the accuracy of MT model. Trainability means CDWs do not cause data fragmentation, and that each CDW unit has sufficiently many examples to train reliable alignments. Trainability is especially critical since any method expanding the translation vocabulary can result into data sparseness problem. However, trainability and consistency can be competing goals to achieve, where a compromise between the two has to be stricken. In this section, we will describe several CDW units satisfying these two criteria to different degrees.

A simple tri-word model \([w_{\text{left}} - w + w_{\text{right}}]\) can take into consideration word \(w\)’s left and right word contexts. Tri-word models, however would clearly lead to data fragmentation simply because there are too many of them. With this in mind, we introduce context classes, which rely on the fact that many word contexts are similar. Context classes can be defined in a number of ways, such as syntactic and semantic clustering. In fact, they have been widely used in language modeling [14] and other natural language processing applications. Using context classes may mitigate the data fragmentation problem to some extent depending upon the number of context classes used to create the CDWs.

Nowadays typical MT systems use anywhere from 20K to more than 100M parallel sentence pairs. Given the vast quantities of data, using CDWs provides the flexibility to model different senses of a word using different surface forms. CDWs also help to reduce the morphological gap between a language pair, if there is an inflectional gap between the source and target language. We can associate a context label to each word based on the mentioned information sources. The context label shows that the words belonging to this class share the same properties. At this point there are several ways the CDWs can be described. It is better to illustrate the idea using an example sentence, “thanks for the help”, in Table 1. The second raw in Table 1 shows the POS tags for the words of the sentence. The first CDW unit is constructed by attaching the POS tags to the words, which is denoted by \(word=\text{pos}\). The second CDW unit is generated by attaching the POS tag of the left context word, \(\text{leftctx}\)\(-\text{word}=\text{pos}\). We also included sentence-begin (SB) and sentence-end (SE) marks as additional POS tags. The third CDW is defined similarly, but by including the POS tag of the right context word, \(word=\text{pos}+\text{rightctx}\). The last row shows the most detailed CDW unit that includes the POS tags of the word and its left and right neighbors, \(leftctx\)\(-\text{word}=\text{pos}+\text{righttext}\).

The tokenized data can be used in two ways in the statistical MT (SMT). First, it can be used as a language model in a lattice/Nbest list rescoring scheme. In other words, the translation model can be built using baseline phrasal units. Then the output of the baseline model (lattice/Nbest) can be tokenized into CDW units. A language model built on the tokenized data can be used to re-rank the lattice/Nbest for improved translation output. The CDW language model can also be interpolated with the baseline word language model.

The second method involves tokenizing the baseline data into CDW data, and building the entire translation model (alignment, phrase tables) using the CDWs. Here in order to reduce the effects data fragmentation on alignments one can build the alignments using the baseline word model and use it for the CDW based phrase extraction.

3. STATISTICAL MACHINE TRANSLATION ARCHITECTURE

The statistical MT problem has been formulated as that of finding the most likely word sequence, \(\hat{c}\), in some target language \(E\), given the word sequence, \(f\), in the source language \(F\) [14]:

\[
\hat{c} = \arg \max_e P(f|e)P(e),
\]

where \(P(e)\) is the language model of \(E\), \(P(e|f)\) is the translation model and the argmax operation denotes the search problem. Hence, a statistical MT system consists of a training phase to construct the translation and language models and a search phase to decode the most likely word sequence in a target language.

Starting from a collection of parallel sentences, we trained word alignment models in two translation directions, from English to Iraqi Arabic and from Iraqi Arabic to English, and derived two sets of Viterbi alignments. By combining word alignments in two directions using heuristics [3], a single set of static word alignments was then formed. All phrase pairs with respect to the word alignment boundary constraint were identified.
and pooled together to build phrase translation tables with the Maximum Likelihood criterion. The maximum number of words in Arabic phrases was set to 5. Our decoder is a phrase-based multi-stack implementation of log-linear models similar to Pharaoh [10]. Like most other Maximum Entropy based decoders, active features in our decoder include translation models in two directions, lexicon weights in two directions, language model, distortion model, and sentence length penalty.

4. EXPERIMENTAL RESULTS and DISCUSSION

We used a parallel corpus of 590K utterance pairs with 109K words (59K morphemes) on the Iraqi Arabic side and 30K words on the English side to train the MT models. This data is collected for a limited domain speech-to-speech translation project [16]. English language model training data uses the English side of the parallel corpora. Since we do not have POS tagged data available on the Iraqi side we applied the proposed technique only in the Iraqi Arabic → English direction. We used SVMTool [12], which reports a tagging accuracy of around 97% using 35 tags on the WSJ corpus, to tag the English data. We neither trained nor tuned the POS tagger using our data. As such, we believe any tagger achieving similar performance figures can be used for our application. Statistical trigram language models with Modified Knesser-Ney smoothing [13] are built using both context independent word (CIW) based data and CDW based tokenized data.

As one can imagine associating three POS tag dimensions with each word can easily result in data fragmentation. Therefore, we defined a stoplist of 150 (empirically determined) most frequent words and defined a single common POS tag for all of them. The total number of word tokens in the training data is 4.7M. Sum of the the word tokens for the top 150 most common words is 3.1M. Therefore, CDWs are generated for the remaining 1.6M tokens.

We used a development set to tune the translation parameters and a test set to measure the final performance. Both development set and test set has about 3.5K utterances. We measure translation performance by the BLEU score [5] with one reference for each hypothesis. In order to evaluate the performance of the CDW based language models, a translation N-best list (N=10) is generated using the baseline language model. First, on the development data all feature weights including the language model weights are optimized to maximize the BLEU score using the downhill simplex method [1]. These weights are fixed when language models are used on the test data. The translation BLEU scores for different language models are given in Table 2. In the upper part of the table individual language model performances are given. The numbers inside the parenthesis in the first column denote the vocabulary size for the respective language models. Introducing the context dependency in the form of word = pos increase the vocabulary size from 30K to 44K. Introducing additional left and right context increased the vocabulary to 102K and 105K, respectively. Using both left and right context in addition to the POS tag increased the vocabulary to 195K. In the last row plain word based baseline language model is interpolated with the POS tag based language model that is built only on the POS sequence (without words). The entries on the middle column are for the development set (devset) with tuned weights that maximized the BLEU scores. Using the word = pos tokens for language modeling improved the result by 0.44 points, and using leftctx-word=pos, word=pos+rightctx and leftctx-word=pos+rightctx improved the scores by about 0.5 points on the development data. On the test data the corresponding improvements are about 0.9, 0.9, 1.0 and 1.0, respectively.

In the lower part of Table 2 we show the N-best rescoring results using interpolated language models. The results both on the development and test data (testset) show that using the baseline trigram language model does not improve the performance, except for leftctx-word=pos+rightctx, which achieves an additional 0.2 points when interpolated with the baseline language model. Using a language model trained on the POS sequence does not improve the results on the development data, but tuning it on the test data gives only a marginal improvement (last row). This is confirming the previous findings that POS sequence seems to be a weak knowledge source by itself in helping the word ordering in MT systems[7, 6].

We also obtained some preliminary results using CDWs in the entire translation process. Table 4 shows the results using word=pos and leftctx-word=pos. We also used the alignments of the baseline model to extract phrases. Building the translation models with word=pos does not result in worse performance compared to the

| Sentence:  | thanks | for | the | help |
| POS tags  | word=pos | leftctx-word=pos+rightctx | word=pos+rightctx | word=pos+rightctx |
| thanks  | NNS | for=IN | the=DT | help=NNS |
| thanks=NNS | for=IN | the=DT | help=NNS |
| SB-thanks=NNS | NNS-for=IN | IN-the=DT | DT-help=NNS |
| SB-thanks=NNS | NNS-for=IN | IN-the=DT | DT-help=NNS |

Table 1: Definition of CDWs as the translation units.
The alignment information to identify the words that will focus on more clever selection of CDWs by us-
potential for further improvements. Hence, our future
MT rescoring scheme. We believe CDW modeling have
fity and trainability criteria. We demonstrated the ef-
f classes. We used POS tags to define context classes
hand, suffers from the "word sense ambiguity" problem.
causes trainability problems. Using CIWs, on the other
means, using larger units in the form of phrases would
capture some of the important effects; however, as the
phrase length grows, they also grow in number, which
causes trainability problems. Using CIWs, on the other
hands, suffers from the "word sense ambiguity" problem.
CDWs are a compromise between specificity and train-
ability. A key issue for CDWs is how to define context
classes. We used POS tags to define context classes and
posed four CDWs that use different degrees of
context information in order to accommodate for speci-
ity and trainability criteria. We demonstrated the ef-
effectiveness of the CDWs using language models in an
MT rescoring scheme. We believe CDW modeling have
potential for further improvements. Hence, our future
work will focus on more clever selection of CDWs by us-
ing the alignment information to identify the words that

5. CONCLUSIONS

Machine translation is a complex process where
words in different parts of a sentence are correlated.
Because most interdependencies between words are lo-
cal, using larger units in the form of phrases would
have multiple senses and apply the context dependent
modeling selectively.

References

[1] F. J. Och and H. Ney, "Discriminative training and maximum
entropy models for statistical machine translation", Proc. ACL-02,

to statistical machine translation", Computational Linguistics,

Statistical Alignment Models," Comp. Linguistics, 29(1):9-51,
2003.

els for Speaker -Independent Continuous Speech Recognition”,

for Automatic Evaluation of Machine Translation”, Proc. ACL,
2002, Philadelphia, PA.

[6] N. Ueffing and H. Ney, "Using POS Information for Statisti-
cal Machine Translation into Morphologically Rich Languages."


[8] M. Popovic and H. Ney, "POS-based Word Reorderings for Sta-


decoder for phrase based statistical machine translation models",


Generator Based on Support Vector Machines", In Proc. LRRC-


[14] P. F. Brown, et. al., "The Mathematics of Statistical Ma-
tchine Translation: Parameter Estimation", Comp. Linguistics,


[16] Y. Gao, et. al., “IBM MASTOR: Multilingual Automatic Speech-

 Morphology Gap for Arabic Statistical Machine Translation”,


<table>
<thead>
<tr>
<th>LM</th>
<th>Arabic → English</th>
<th>DevSet</th>
<th>TestSet</th>
</tr>
</thead>
<tbody>
<tr>
<td>Baseline (30K)</td>
<td>0.4675</td>
<td>0.4715</td>
<td></td>
</tr>
<tr>
<td>word=pos (44K)</td>
<td>0.4717</td>
<td>0.4802</td>
<td></td>
</tr>
<tr>
<td>leftctx-word=pos (162K)</td>
<td>0.4726</td>
<td>0.4800</td>
<td></td>
</tr>
<tr>
<td>leftctx-word=pos+rightctx (195K)</td>
<td>0.4729</td>
<td>0.4811</td>
<td></td>
</tr>
<tr>
<td>Baseline + word=pos</td>
<td>0.4717</td>
<td>0.4809</td>
<td></td>
</tr>
<tr>
<td>Baseline + leftctx-word=pos</td>
<td>0.4720</td>
<td>0.4813</td>
<td></td>
</tr>
<tr>
<td>Baseline + word=pos+rightctx</td>
<td>0.4731</td>
<td>0.4826</td>
<td></td>
</tr>
<tr>
<td>Baseline + rightctx</td>
<td>0.4723</td>
<td>0.4725</td>
<td></td>
</tr>
</tbody>
</table>

Table 3: Sample translation outputs with Baseline model rescored with leftctx-word=pos language model.

<table>
<thead>
<tr>
<th>LM</th>
<th>Arabic → English</th>
<th>Own Alignm’t</th>
<th>Baseline</th>
<th>Own Alignm’t</th>
<th>Baseline</th>
</tr>
</thead>
<tbody>
<tr>
<td>Baseline</td>
<td>0.4673</td>
<td>0.4673</td>
<td>0.4715</td>
<td>0.4715</td>
<td></td>
</tr>
<tr>
<td>word=pos</td>
<td>0.4690</td>
<td>0.4682</td>
<td>0.4725</td>
<td>0.4742</td>
<td></td>
</tr>
<tr>
<td>leftctx-word=pos</td>
<td>0.4648</td>
<td>0.4677</td>
<td>0.4696</td>
<td>0.4733</td>
<td></td>
</tr>
</tbody>
</table>

Table 4: Using CDWs as translation units.