Lexical Selection for Hybrid MT with Sequence Labeling

Alex Rudnick and Michael Gasser
Indiana University, School of Informatics and Computing
{alexr,gasser}@indiana.edu

Abstract

We present initial work on an inexpensive approach for building large-vocabulary lexical selection modules for hybrid RBMT systems by framing lexical selection as a sequence labeling problem. We submit that Maximum Entropy Markov Models (MEMMs) are a sensible formalism for this problem, due to their ability to take into account many features of the source text, and show how we can build a combination MEMM/HMM system that allows MT system implementors flexibility regarding which words have their lexical choices modeled with classifiers. We present initial results showing successful use of this system both in translating English to Spanish and Spanish to Guarani.

1 Introduction

Lexical ambiguity presents a serious challenge for rule-based machine translation (RBMT) systems, since many words have several possible translations in a given target language, and more than one of them may be syntactically valid in context. A translation system must choose a translation for each word or phrase in the input sentence, and simply taking the most common translation will often fail, as a word in the source language may have translations in the target language with significantly different meanings. Even when choosing among near-synonyms, we would like to respect selectional preferences and common collocations to produce natural-sounding output text.

Writing lexical selection rules by hand is tedious and error-prone; even if informants familiar with both languages are available, they may not be able to enumerate the contexts under which they would choose one translation alternative over another. Thus we would like to learn from corpora where possible.

Framing the resolution of lexical ambiguities in machine translation as an explicit classification task has a long history, dating back at least to early SMT work at IBM (Brown et al., 1991). More recently, Carpuat and Wu have shown how to use word-sense disambiguation techniques to improve modern phrase-based SMT systems (Carpuat and Wu, 2007), even though the language model and phrase tables of these systems can mitigate the problem of lexical ambiguities somewhat. Treating lexical selection as a word-sense disambiguation problem, in which the sense inventory for each source-language word is its set of possible translations, is often called cross-lingual WSD (CL-WSD). This framing has received enough attention to warrant shared tasks at recent SemEval workshops; the most recent running of the task is described in (Lefever and Hoste, 2013).

Intuitively, machine translation implies an "all-words" WSD task: we need to choose a translation for every word or phrase in the source sentence, and the sequence of translations should make sense taken together. Here we begin to explore CL-WSD not just as a classification task, but as one of sequence labeling. We describe our approach and implementation, and present two experiments. In the first experiment, we apply the system to the SemEval 2013 shared task on CL-WSD (Lefever and Hoste, 2013), translating from English to Spanish, and in the second, we perform an all-words labeling task, translating text from the Bible from Spanish to Guarani. This is work in progress and our code is currently "research-quality", but we are developing the software in the open[1], with the intention of using it with free RBMT systems and producing an easily reusable package as the system matures.

[1] http://github.com/alexrudnick/clwsd

2 Related Work

To our knowledge, there has not been work specifically on sequence labeling applied to lexical selection for RBMT systems. However, there has been work recently on using WSD techniques for translation into lower-resourced languages, such as the English-Slovene language pair, as in (Vintar et al., 2012).

The Apertium team has a particular practical interest in improving lexical selection in RBMT; they recently have been developing a new system, described in (Tyers et al., 2012), that learns finite-state transducers for lexical selection from the available parallel corpora. It is intended to be both very fast, for use in practical translation systems, and to produce lexical selection rules that are understandable and modifiable by humans.

Outside of the CL-WSD setting, there has been work on framing all-words WSD as a sequence labeling problem. Particularly, Molina *et al.* (2002) have made use of HMMs for all-words WSD in a monolingual setting.

3 Sequence Labeling with HMMs

In building a sequence-based CL-WSD system, we first tried using the familiar HMM formalism. An HMM is a generative model, giving us a formula for $P(S,T) = P(T) * P(S|T)$. Here by S we mean a sequence of source-language words, and by T we mean a sequence words or phrases in the target language. In practice, the input sequence S is a given, and we want to find the sequence T that maximizes the joint probability, which means predicting an appropriate label for each word in the input sequence.

Using the (first-order) Markov assumption, we approximate $P(T)$ as $P(T) = \prod_i P(t_i|t_{i-1})$, where i denotes each index in the input sentence. Then we imagine that each source-language word s_i is generated by the corresponding unobserved label t_i, through the emission probabilities $P(s|t)$. This generative model is admittedly less intuitive for CL-WSD than for POS-tagging (where it is more traditionally applied), in that it requires the target-language words to be generated in the source order.

Training the transition model – roughly an n-gram language model – for target-language words or phrases in the source order is straightforward with sentence-aligned bitext. We use one-to-many alignments in which each source word corresponds with zero or more target-language words, and we take the sequence of target-language words aligned with a given source word to be its label. NULL labels are common; if a source word is not aligned to a target word, it gets a NULL label. Similarly , we can learn the emission probabilities, $P(s|t)$, simply by counting which source words are paired with which target words and smoothing.

For decoding with this model, we can use the Viterbi algorithm, especially for a first-order Markov model – although we must be careful in the inner loops only to consider the possible target-language words and not the entire target-language vocabulary. The Viterbi algorithm may still be used with second- or higher-order models, although it slows down considerably. In the interest of speed, in this work we performed decoding for second-order HMMs with a beam search.

4 Sequence Labeling With MEMMs and HMMs

Contrastingly, an MEMM is a discriminative sequence model, with which we can calculate the conditional probability $P(T|S)$ using individual discriminative classifiers that model $P(t_i|F)$ (for some features F). Like an HMM, an MEMM models transitions over labels, although the input sequence is considered given. This frees us to include any features we like from the source-language sentence. The "Markov" aspect of the MEMM is that, unlike a standard maximum entropy classifier, we can include information from the previous k labels as features, for a k-th order MEMM. So at every step in the sequence labeling, we want a classifier that models $P(t_i|S, t_{i-1}...t_{i-k})$, and the probability of a sequence T is just the product of each of the individual transition probabilities.

To avoid the intractable task of building a single classifier that might return thousands of different labels, we could in principle build a classifier for each individual word in the source-language vocabulary, each of which will produce perhaps tens of possible target-language labels. However, there will be tens or hundreds of thousands of words in the source-language vocabulary, and most word-types will only occur very rarely; it may be prohibitively expensive to train and store classifiers for each of them.

We would like a way to focus our efforts on some words, but not all, and to back off to a simpler model when a classifier is not available for a given word. Here, in order to approximate $P(t_i|S, t_{i-1}...t_{i-k})$, we use an HMM, as described in the previous section, with which we can estimate $P(s_i, t_i|t_{i-1}...t_{i-k})$ as $P(t_i|t_{i-1}...t_{i-k}) * P(s_i|t_i)$. This gives us the joint probability, which we divide by $P(s_i)$ – prior probabilities of each source-language word must be stored ahead of time – and thus we can approximate the conditional probability that we need to continue the sequence labeling.

In the implementation, we can specify criteria under which a source-language word will have its translations explicitly modeled with a maximum entropy classifier. When training a system, one might choose, for example, the 100 most common ambiguous words, all words that are observed a certain number of times in the training corpus, or words that are particularly of interest for some other reason.

At training time, we find all of the instances of the words that we want to model with classifiers, along with their contexts, so that we can extract appropriate features for training the classifiers. Then we train classifiers for those words, and store the classifiers in a database for retrieval at inference time.

For inference with this model, we use a beam search rather than the Viterbi algorithm, for convenience and speed while using a second-order Markov model. A sketch of the beam search implementation is presented in Figure 1.

5 Experiments

So far, we have evaluated our sequence-labeling system in two different settings, the English-Spanish subset of a recent SemEval shared task (Lefever and Hoste, 2013), and an all-words prediction task in which we want to translate, from Spanish to Guarani, each word in a test set sampled from the Bible.

5.1 SemEval CL-WSD task

In the SemEval CL-WSD task, systems must provide translations for twenty ambiguous English nouns given a small amount of context, typically a single sentence. The test set for this task consists of fifty short passages for each ambiguous word, for a thousand test instances in total. Each passage contains one or a few uses of the ambiguous word. For each test passage, the system must produce a translation of the noun of interest into the target language. These translations may be a single word or a short phrase in the target language, and they should be lemmatized. The task allows systems to produce several output labels, although the scoring metric encourages producing one best guess, which is matched against several reference translations provided by human annotators. The details of the scoring are provided in the task description paper, and the scores reported were calculated with a script provided by the task organizers.

As a concrete example, consider the following sentences from the test set:

(1) But a quick look at today's *letters* to the editor in the Times suggest that here at least is one department of the paper that could use a little more fact-checking.

(2) All over the ice were little Cohens, little Levys, their names sewed in block *letters* on the backs of their jerseys.

A system should produce *carta* (a message or document) for Sentence (1) and *letra* or *carácter* (a symbol or handwriting) for (2). During sequence labeling, our system chooses a translation for each word in the sentence, but the scoring only takes into account the translations for the words marked in italics.

For simplicity and comparability with previous work, we trained our system on the Europarl Intersection corpus, which was provided for developing CL-WSD systems in the shared task. The Europarl Intersection is a subset of the sentences from Europarl (Koehn, 2005) that are available in English and all five of the target languages for the task, although for these initial experiments, we only worked with Spanish. There were 884603 sentences in our training corpus.

We preprocess the Europarl training data by tokenizing with the default NLTK tokenizer (Bird et al., 2009), getting part-of-speech tags for the English text with the Stanford Tagger (Toutanova et al., 2003), and lemmatizing both sides with TreeTagger (Schmid, 1995). We aligned the untagged English text with the Spanish text using the Berkeley Aligner (DeNero and Klein, 2007) to get one-to-many alignments from English to Spanish, since the target-language labels in this setting may be multi-word phrases. We used nearly the default settings for Berkeley Aligner, except that we

```
def beam_search(sequence, HMM, source_word_priors, classifiers):
    """Search over possible label sequences, return the best one we find."""
    candidates = [Candidate([], 0)] # empty label sequence with 0 penalty
    for t in range(len(sequence)):
        sourceword = sequence[t]
        for candidate in candidates:
            context = candidate.get_context(t) # labels at positions (t−2, t−1)
            if sourceword in classifiers:
                features = extract_features(sequence, t, context)
                label_distribution = classifiers[sourceword].prob_classify(features)
            else:
                label_distribution = Distribution()
                for label in get_vocabulary(sourceword):
                    label_distribution[label] = (HMM.transition(context, label) +
                                                 HMM.emission(sourceword, label) −
                                                 source_word_priors[sourceword])
            # extend candidates for next time step to include labels for next word
            add_new_candidates(candidate, label_distribution, new_candidates)
        candidates = filter_top_k(new_candidates, BEAMWIDTH)
    return get_best(candidates)
```

Figure 1: Python-style code sketch for MEMM/HMM beam search. Here we are using negative log-probabilities, which we interpret as penalties to be minimized.

ran 20 iterations each of IBM Model 1 and HMM alignment.

We trained classifiers for all of the test words, and also for any words that appear more than 500 times in the corpus. The classifiers used the previous two labels and all of the tagged, lemmatized words within three tokens on either side of the target word as features. Training was done with the MEGA Model optimization package [2] and its corresponding NLTK interface.

At testing time, for each test instance, we labeled the test sentences with four different sequence labeling methods: first-order HMMs, second-order HMMs, MaxEnt classifiers with no sequence features, and the MEMMs with HMM backoff. We then compared the system output against the reference translations for the target words using the script provided by the task organizers.

5.2 All-words Lexical Selection for Spanish-Guarani

Since we are primarily interested in lexical selection for RBMT systems in lower-resource settings, we also experimented with translating from Spanish to Guarani, using the Bible as bitext. In this experiment, we labeled all of the text in the test set using each of the different sequence labeling models, and we report the classification accuracy over the test set.

For example, for the following sentences – from Isaiah and Psalms, respectively – the system should predict the corresponding Guarani roots for each Spanish word. Here we show the inflected Spanish and Guarani text with English translation for the sake of readability, although the system was given the roots of the Spanish words as produced by the morphological analyzer.

(3) a. Plantaréis viñas y *comeréis* su fruto.
b. Peñotỹ parral ha *pe'u* hi'a.
c. You will plant vineyards and *eat* their fruit.

(4) a. *Comieron* y se saciaron.
b. *Okaru* hikuái hyguãtã meve.
c. They *ate* and were well filled.

In this example, the correct translation of *comer* depends on transitivity: if transitive, it should be an inflected form of *'u* as in (3), if intransitive it should be *karu*, as in (4).

In preparing the corpus, since different translations of the Bible do not necessarily have direct correspondences between verse numbers (they are not unique identifiers across language!), we selected only the chapters that contain the same number of verses in our Spanish and Guarani translations. This only leaves 879 chapters out of 1189, for a total of 22828 bitext verses of roughly one sentence each. We randomly sampled 100 verses from the corpus and set these aside as the test set.

[2] http://www.umiacs.umd.edu/~hal/megam/

Here we trained the HMM and MEMM as before, but with lemmatized Spanish as the source language, and the roots of Guarani words as the target. As Guarani is a much more morphologically rich language than either English or Spanish, this requires the use of a sophisticated morphological analyzer, described in section 6. Due to the much smaller data set, in this setting we stored classifiers for any Spanish word that occurs more than 20 times in the training data and backed off to the HMM during decoding otherwise.

6 Morphological Analysis for Guarani

We analyze the Spanish and Guarani Bible using our in-house morphological analyzer, originally developed for Ethiopian Semitic languages (Gasser, 2009). As in other, more familiar, modern morphological analyzers such as (Beesley and Karttunen, 2003), analysis in our system is modeled by cascades of finite-state transducers (FSTs). To solve the problem of long-distance dependencies, we extend the basic FST framework using an idea introduced by Amtrup (2003). Amtrup starts with the well-understood framework of weighted FSTs, familiar from speech recognition. For speech recognition, FST arcs are weighted with probabilities, and a successful traversal of a path through a transducer results in a probability that is the product of the probabilities on the arcs that are traversed, as well as an output string as in conventional transducers. Amtrup showed that probabilities could be replaced by feature structures and multiplication by unification. In an FST weighted with feature structures, the result of a successful traversal is the unification of the feature structure "weights" on the traversed arcs, as well as an output string. Because a feature structure is accumulated during the process of transduction, the transducer retains a sort of memory of where it has been, permitting the incorporation of long-distance constraints such as those relating the negative prefix and suffix of Guarani verbs.

In our system, the output of the morphological analysis of a word is a root and a feature structure representing the grammatical features of the word. We implemented separate FSTs for Spanish verbs, for Guarani nouns, and for the two main categories of Guarani verbs and adjectives. Since Spanish nouns and adjectives have very few forms, we simply list the alternatives in the lexicon for these categories. For this paper, we are only concerned with the roots of words in our corpora, so we ignore the grammatical features that are output with each word.

7 Results

The scores for the first experiment are presented in Figure 2. Here we use the precision metric calculated by the scripts for the SemEval shared task (Lefever and Hoste, 2013), which compare the answers produced by the system against several reference answers given by human annotators. There are two "most frequent sense" baselines reported. The first one ("with tag"), is the baseline in which we always take the most frequent label for a given source word, conditioned on its POS tag. The other MFS baseline is not conditioned on POS tag; this was the baseline for the SemEval task. Perhaps unsurprisingly, we see part-of-speech tagging doing some of the lexical disambiguation work.

Neither of the HMM systems beat the most-frequent-sense baselines, but both the non-sequence MaxEnt classifier and the MEMM system did, which suggests that the window features are useful in selecting target-language words. Furthermore, the MEMM system outperforms the MaxEnt classifiers.

The scores for the second experiment are presented in Figure 3. Here we did not have human-annotated reference translations for each word, so we take the labels extracted from the alignments as ground truth and can only report per-word classification accuracy, rather than the more sophisticated precision metric used in the shared task.

Here we see similar results. Neither of the HMM systems beat the MFS baseline, and the trigram model was noticeably worse. The training set here is probably too sparse to train a good trigram model. The MEMM system, however, did beat the baseline, posting the highest results: just over two-thirds of the time, we were able to predict the correct label for each Spanish word, whereas the most frequent label was correct about 60% of the time.

8 Conclusions and Future Work

We have described a work-in-progress lexical selection system that takes a sequence labeling approach, and shown some initial successes in using it for cross-language word sense disambiguation tasks for English to Spanish and Spanish to Guarani. We have demonstrated a hybrid se-

system	features	score (precision)
MFS (with tag)		24.97
MFS (without tag)		23.23
HMM1	current word, previous label	21.17
HMM2	current word, previous two labels	21.23
MaxEnt	three-word window	25.64
MEMM	three-word window, previous two labels	**26.49**

Figure 2: Results for the first experiment; SemEval 2013 CL-WSD task.

system	features	score (accuracy %)
MFS		60.39
HMM1	current word, previous label	57.40
HMM2	current word, previous two labels	43.04
MEMM	three-word window, previous two labels	**66.82**

Figure 3: Results for the second experiment; all-words lexical selection on the Guarani Bible

quence labeling strategy that combines MEMMs and HMMs, which will allow users to set parameters sensibly for their computational resources and available training data.

In future work, we will continue to refine the approach, exploring different parameter settings, such as beam widths, numbers of classifiers for the MEMM component, and the effects of different features as input to the classifiers. We are also interested in making use of multilingual information sources, as in the work of Lefever and Hoste (2011). We may also consider more sophisticated sequence tagging models, such as CRFs (Lafferty et al., 2001), although we may not have enough training data to make use of richer models.

Our goal for this work is practical; we are trying to produce a hybrid Spanish-Guarani MT system that can be used in Paraguay. We have a small amount of Guarani training data available, and plan to collect more. At the time of writing, our lexical selection system is a prototype and not yet integrated with our RBMT engine, but this integration is among our near-term goals.

A limitation of the current design is that we do not yet have a good way to make use of monolingual training data. In SMT, it is common practice to train a language model for the target language from a monolingual corpus that is much larger than the available bitext. There is a substantial amount of available Guarani text on the Web, but our current model can only be trained on aligned bitext. Given Guarani text that had been rearranged into a Spanish-like word order, we could build a better model for the transition probabilities in the HMM component of the system. It might be feasible to use a Guarani-language parser and some linguistic knowledge for this purpose. We will also investigate ways to translate multiword expressions as a unit rather than word-by-word.

References

Jan Amtrup. 2003. Morphology in machine translation systems: Efficient integration of finite state transducers and feature structure descriptions. *Machine Translation*, 18.

Kenneth R. Beesley and Lauri Karttunen. 2003. *Finite State Morphology*. CSLI Publications.

Steven Bird, Ewan Klein, and Edward Loper. 2009. *Natural Language Processing with Python*. O'Reilly Media.

Peter F. Brown, Stephen A. Della Pietra, Vincent J. Della Pietra, and Robert L. Mercer. 1991. Word-Sense Disambiguation Using Statistical Methods. In *Proceedings of the 29th Annual Meeting of the Association for Computational Linguistics*.

Marine Carpuat and Dekai Wu. 2007. How Phrase Sense Disambiguation Outperforms Word Sense Disambiguation for Statistical Machine Translation. In *11th Conference on Theoretical and Methodological Issues in Machine Translation*.

John DeNero and Dan Klein. 2007. Tailoring Word Alignments to Syntactic Machine Translation. In *Proceedings of the 45th Annual Meeting of the Association of Computational Linguistics*. Association for Computational Linguistics, June.

Michael Gasser. 2009. Semitic morphological analysis and generation using finite state transducers with feature structures. In *Proceedings of the 12th Conference of the European Chapter of the ACL*.

Philipp Koehn. 2005. Europarl: A Parallel Corpus for Statistical Machine Translation. In *Proceedings of The Tenth Machine Translation Summit*.

John D. Lafferty, Andrew McCallum, and Fernando C. N. Pereira. 2001. Conditional Random Fields: Probabilistic Models for Segmenting and Labeling Sequence Data. In *ICML*.

Els Lefever and Véronique Hoste. 2013. SemEval-2013 Task 10: Cross-Lingual Word Sense Disambiguation. In *Proceedings of the 7th International Workshop on Semantic Evaluation (SemEval 2013)*.

Els Lefever, Véronique Hoste, and Martine De Cock. 2011. ParaSense or How to Use Parallel Corpora for Word Sense Disambiguation. In *Proceedings of the 49th Annual Meeting of the Association for Computational Linguistics: Human Language Technologies*.

Antonio Molina, Ferran Pla, and Encarna Segarra. 2002. A Hidden Markov Model Approach to Word Sense Disambiguation. In *IBERAMIA*.

Helmut Schmid. 1995. Improvements In Part-of-Speech Tagging With an Application To German. In *Proceedings of the ACL SIGDAT-Workshop*.

Kristina Toutanova, Dan Klein, Christopher D. Manning, and Yoram Singer. 2003. Feature-Rich Part-of-Speech Tagging with a Cyclic Dependency Network. In *PROCEEDINGS OF HLT-NAACL*.

F. M. Tyers, F. Sánchez-Martínez, and M. L. Forcada. 2012. Flexible finite-state lexical selection for rule-based machine translation. In *Proceedings of the 17th Annual Conference of the European Association of Machine Translation, EAMT12*.

Špela Vintar, Darja Fišer, and Aljoša Vrščaj. 2012. Were the clocks striking or surprising? Using WSD to improve MT performance. In *Proceedings of the Joint Workshop on Exploiting Synergies between Information Retrieval and Machine Translation (ESIRMT) and Hybrid Approaches to Machine Translation (HyTra)*.