SYNERGY: A Named Entity Recognition System for Resource-scarce Languages such as Swahili using Online Machine Translation

Rushin Shah, Bo Lin, Anatole Gershman, Robert Frederking

Language Technologies Institute, School of Computer Science, Carnegie Mellon University
5000 Forbes Avenue, Pittsburgh 15213, PA, USA
{rnshah, bolin, anatoleg, ref+}@cs.cmu.edu

Abstract

Developing Named Entity Recognition (NER) for a new language using standard techniques requires collecting and annotating large training resources, which is costly and time-consuming. Consequently, for many widely spoken languages such as Swahili, there are no freely available NER systems. We present here a new technique to perform NER for new languages using online machine translation systems. Swahili text is translated to English, the best off-the-shelf NER systems are applied to the resulting English text and the English named entities are mapped back to words in the Swahili text. Our system, called SYNERGY, addresses the problem of NER for a new language by breaking it into three relatively easier problems: Machine Translation to English, English NER and word alignment between English and the new language. SYNERGY achieves good precision as well as recall for Swahili. We also apply SYNERGY to Arabic, for which freely available NERs do exist, in order to compare its performance to other NERs. We find that SYNERGY's performance is close to the state-of-the-art in Arabic NER, with the advantage of requiring vastly less time and effort to build.

1. Introduction

Online machine translation tools such as (Google Translate™, 2010) and (Microsoft Bing Translator™, 2010) support many languages, including some for which few resources exist and few NLP tools have been developed. In this paper, we focus on one such resource-scarce language, Swahili. As of August 2009, there is no freely available Swahili NER system (Borovikov et al., 2009). However, Google Translate supports two-way translation between Swahili and English. Admittedly, the quality of translation is far from perfect. Many words are simply not translated from Swahili and are carried over as they are into the English version. However, the output can be leveraged to achieve our purposes.

We will show that by performing NER on the translated English text and then matching back the English named entities to words in the Swahili text, our SYNERGY system can perform Swahili NER with a high degree of accuracy. Moreover, SYNERGY can be easily applied to perform NER for other languages too. This is significant because we no longer need to acquire and annotate large amounts of training data ourselves, which is costly and time-consuming for each new language.

Since there is no other Swahili NER system available, we need to apply SYNERGY to another language, for which freely available NER systems do exist, in order to compare its performance to these systems and evaluate the effectiveness of our Machine Translation (MT) based approach to NER. We use Arabic for this purpose, because it is typically considered a 'hard' language for which to do NER, and hence presents a good test for SYNERGY.

The remainder of this paper is organized as follows. In section 2, we discuss relevant prior work done in this area. Section 3 details the various datasets and tools that we have used. In section 4 we describe the development of our SYNERGY NER architecture, presenting in section 5 the results of this system. In section 6, we present examples that illustrate SYNERGY's performance. Finally, in section 7, we discuss our conclusions and plans for future work in this area.

2. Related Work

The problem of NER can be thought of as a subtask of the general task of information extraction. It involves identifying named entities in a text, and in some cases, classifying them according to various types such as persons, locations, organizations, etc. In this paper, we restrict ourselves to the task of identifying named entities only and do not try to predict their types. The NER task was formally defined at the (MUC6, 1995) conference, and this definition was expanded upon at the (CoNLL, 2002) and (CoNLL, 2003) conferences. Various NER systems were evaluated at these conferences, and NER systems produced in subsequent years have also been evaluated on the (CoNLL, 2003) test set. (Zhao et al., 2007) and (Huang, 2006) have shown that NER can be used to help machine translation; it is important to note that our work focuses on showing the reverse case, which we believe to be novel. (De Pauw et al., 2009) illustrate the difficulties inherent in English-Swahili word alignment during their development of the SAWA English-Swahili parallel corpus. In addition, (De Pauw et al., 2006) have also developed a memory-based part-of-speech (POS) tagger for Swahili, using the Helsinki Corpus of Swahili. (Benajiba and Rosso, 2008) use Conditional Random Fields (CRFs) to perform Arabic NER and report a best-case F1 score of 0.792. (Benajiba et al., 2008) use Support Vector Machines (SVMs) and CRFs with optimized feature sets to perform Arabic NER and report a best-case F1 score of 0.835.

3. Resources

We use (Google Translate™, 2010) as our online Machine Translation system for Swahili and Arabic text. We attempted to use (Microsoft Bing Translator™, 2010) (which

doesn't support Swahili yet) for Arabic but were unable to do so because currently it leaves many words (including many Named Entities) untranslated and this leads to poor NER performance. We would have liked to use additional MT systems, but for these languages, we could find only two freely available machine translation tools. Although various other MT systems are available online, they mainly support translation only between English and European languages. By contrast, Google Translate and Bing Translator not only support a wide variety of languages but are also steadily adding support for new languages over time. Currently, no other African languages are supported by these or any other freely available MT systems that we know of, but a key advantage of SYNERGY is that as soon as new languages become available on any MT system, SYNERGY can be used with minimal modification to perform NER for these new languages. SYNERGY does not necessarily have to use online MT systems, it can also use MT systems automatically generated from parallel data using freely available toolkits such as (Moses, 2010).

Named Entity (NE) labeled data is scarce for both languages. For Swahili, we use a test set of slightly more than 27000 tokens from the Helsinki Corpus of Swahili (HCS, 2004). For Arabic, we use a test set of 25000 tokens from the ANERcorp corpus (Benajiba et al., 2007). For Named Entity recognition, we use two well-known systems: Stanford's Conditional Random Field (CRF) based NER system (Finkel et al., 2005) and UIUC's Learning-Based Java (LBJ) Named Entity Tagger (Ratinov and Roth, 2009). We use only off-the-shelf named entity recognizers, to demonstrate that a good NER system can be developed for new languages in short order, with minimal training data. We also use the (CoNLL, 2003) dataset.

Post-processing improves the quality of the system, but is not required for its operation. For post-processing, we make use of the Swahili-English dictionary by (Kamusi Project, 2010), and the Linguistic Data Consortium's Arabic TreeBank (Maamouri et al., 2005).

4. SYNERGY Architecture

As described in our introduction, SYNERGY performs NER on a new language, which we shall henceforth refer to as the source language, by translating source language text to English, running off-the-shelf state-of-the-art NER systems on the translated English text, and then matching back the English named entities to words in the source language. It also performs some post-processing to improve NER performance. This section describes each of these steps in detail.

4.1. Translation to English

SYNERGY uses a Perl interface to Google Translate. It translates Swahili and Arabic documents to English by translating each sentence individually. Unlike its web interface, the Google Translate API does not accept sentences that are longer than a certain length (which in our testing we found to be 700 characters). Most corpora, including the ones we use, are composed of newswire articles, where sentence lengths frequently exceed that number. Therefore, when the system encounter sentences whose length exceeds the maximum amount, it performs preprocessing and splits them into smaller sentences of acceptable size. Although splitting sentences in this manner may consequently lead to poor translation results, we find that named entities are not affected and hence this does not present a problem for our system. We also observe that the Arabic translation is of a better quality than the Swahili translation; in particular, many Swahili words are simply not recognized and hence not translated. However, named entities are not strongly affected by this issue either. Let d_{src} denote a source language document; d_{en} denotes its translated English version.

4.2. NER on Translated English Text

For English NER, the freely available systems developed by Stanford and UIUC achieve a high degree of accuracy and represent the current state-of-the-art in the field. We use both these systems in SYNERGY. We initially also tested the (LingPipe™, 2010) and (AlchemyAPI™, 2010) NER systems, but found no improvement in precision or recall from adding either of these systems to SYNERGY.

The Stanford and UIUC systems do not return identical results: Approximately 3% of the named entities are correctly identified by only one of the two systems. We exploit this discrepancy and create a combined NER system called Union that achieves a higher F1 score than either of the individual systems. Formally, the Union system performs NER as follows: A token t is categorized as being part of a named entity if it is classified as being part of a named entity by either the Stanford or UIUC system. We define the system in this manner to try to maximize recall, with the expectation that the increase in recall will compensate for a drop in precision. The Stanford and UIUC systems have their own distinct tokenizers, and there are many subtle differences in the way they treat non-word characters, dates, URLs, etc. So, synchronizing the outputs of the two systems proves to be a non-trivial task.

Table 1 gives a summary of the performance of these systems. The systems were not retrained; we used their off-the-shelf versions. The systems were evaluated on the test subset of the (CoNLL, 2003) dataset. We performed the evaluation on a per-token basis, as opposed to a per-entity basis. This eliminates the need to address potential ambiguities that may be caused if a token is classified as being a part of different named entities by different NER systems. As expected, we find that the Union system gets a slightly higher F1 score than both its component systems. Therefore, Union is the default NER module used in SYNERGY.

System	Precision	Recall	F1
Stanford	0.962	0.963	0.963
UIUC	0.968	0.964	0.966
LingPipe	0.624	0.554	0.587
AlchemyAPI	0.680	0.506	0.580
Union	0.952	0.985	0.968

Table 1: Performance of various NER systems

We run the Union NER system on the translated English document d_{en} and extract a list L_{ne} of named entities and the locations at which they occur in d_{en}.

4.3. Alignment of Named Entities with Source Language Words

This is the final and most challenging operation performed by the SYNERGY system. After creating the list L_{ne} of named entities in the translated English document d_{en}, we need to find the words in the source language document d_{src} that match the entities in L_{ne}. We try two different algorithms: brute force alignment, and word alignment using GIZA++.

4.3.1. Brute Force Alignment

In our initial approach, we translated each word w_{en} in L_{ne} back to a source language word, which we denote as w_{re-src}, and proceeded to search in the source document d_{src} for a word w_{src} that matches w_{re-src}, by looking at a window of words, centered around the location that w_{en} occurs in the translated English document d_{en}. If a word w_{src} matching w_{re-src} was found, it was tagged as being part of a named entity. If no such match was found for this particular w_{re-src} in this window, w_{re-src} was discarded and we repeated this search with the next word in L_{ne}. The parameter in this search algorithm was the size s of the window of words that we examine for each w_{re-src}. We found that this initial approach gave very poor results for both our Swahili and Arabic test sets. It found matches for only 40% of the named entity words in L_{ne}. There was an increase in the number of matches found with an increase in s, until $s = 30$, after which there was no increase. Subsequent error analysis shows that a major cause of this poor performance is that when a source language word w_{src} is translated to an English word w_{en} which is then retranslated to a word w_{re-src} in the original language, it is often the case that $w_{src} \neq w_{re-src}$. For Arabic, additional mismatches are introduced while converting text from UTF-8 to Latin encoding and back.

Hence, we try a different version of this algorithm: After creating L_{ne}, we create a new English document d_{en-tl} by translating each word in the source language document d_{src} one at a time to English. We now search in d_{en-tl} for matches for each word in L_{ne}. To deal with the possibility that a source word w_{src} may produce multiple English words, we keep pointers from each English word to the source word w_{src} that produces it. The search proceeds in the same manner as outlined in the previous algorithm: by considering a window of words of size s centered at the location of the named entity word in d_{en}. If a word w_{en-tl} is found in d_{en-tl} that matches a word in L_{ne}, we label the source language word w_{src} that produces w_{en-tl} as being part of a named entity.

This second version produces much better results and finds matches in the source document d_{src} for 75% of the named entity words. As in the previous algorithm, the number of matches found increases with an increase in s, until $s = 30$. Furthermore, since this algorithm compares English words, we can use stemming to perform a more sophisticated comparison than simple equality testing. We use the well-known (Porter, 1980) stemmer. This augmented version of our brute force alignment algorithm finds matches for 77% of the named entity words. It is important to emphasize that this figure only refers to the fraction of named entity words for which words in the source language are found; it does not address the issue of whether the source language words are in fact the correct ones that produced the named entity words. If many of the matched source language words are incorrect, this results in poor overall NER performance in spite of a large fraction of named entities being matched. Hence, the fraction of matches found is ultimately a metric of limited utility and cannot be reliably construed as a predictor of overall NER performance.

4.3.2. Word Alignment using GIZA++

The GIZA++ package (Och and Hey, 2003) is the state-of-the-art for the task of word alignment in the field of statistical Machine Translation. It contains a number of statistical alignment models specifically designed for word alignment between different languages. We modify SYNERGY to use GIZA++ to perform word alignment. GIZA++ takes an input language corpus and an output language corpus, and automatically generates word classes for both languages using these corpora. It then finds for each sentence in the input language corpus the most probable alignment of the corresponding output language sentence (also known as the Viterbi alignment). A detailed description of the various statistical models used in word alignment and available in GIZA++ can be found in (Och and Hey, 2003). For our task, the translated English documents serve as the input corpus and the source language documents as the output corpus. In order to use GIZA++, SYNERGY first preprocesses these corpora to make sure that they are sentence-aligned.

4.4. Post-Processing

We compile exhaustive part-of-speech tagged English, Swahili and Arabic dictionaries, and divide all of these into Named Entity (NE) and Non-Named Entity (NNE) sections, depending on whether a word may be used as a proper noun or not. Of course, many words may occur in both sections, since in every language many proper nouns are often derived from common words. The post-processing (PP) module scans through the NE-annotated source language documents returned by the previous modules and applies the following two rules:

- If a word annotated as a Named Entity occurs in the NNE section of a dictionary but not in its NE section, the word's annotation is removed.
- If a word not annotated as a Named Entity occurs in the NE section of a dictionary but not in its NNE section, the word is accordingly annotated as a Named Entity.

Even though the above rules are relatively simple, they nevertheless lead to significant improvements in NER performance.

5. SYNERGY Results

Table 2 shows the results of SYNERGY on Swahili data. We use the gold-standard labels in the HCS corpus to check the accuracy of our labeling. Evaluation is performed on a per-token basis, as described earlier. We find

that without post-processing, brute force alignment performs slightly better than GIZA++ based alignment, but with post-processing the situation is reversed and GIZA++ based alignment performs much better. Moreover, post-processing leads to major improvements in the F1 score for both algorithms. In particular, the "GIZA++ with Post-Processing" version of SYNERGY yields an F1 score of 0.815, which we believe is very good for a first effort in the field of Swahili NER.

Version	Prec	Recl	F1
Brute Force w/o PP	0.676	0.694	0.685
Brute Force with PP	0.818	0.704	0.757
GIZA++ w/o PP	0.534	0.900	0.670
GIZA++ with PP	0.754	0.886	0.815

Table 2: SYNERGY Results for Swahili NER

Since there are no other freely available Swahili NER systems, in order to measure the effectiveness of SYNERGY's MT-based approach to NER, we compare SYNERGY's results on Arabic data with the results obtained by state-of-the-art systems in Arabic NER. The results are shown in Table 3. We use the gold-standard labels in ANERcorp to check the accuracy of our labeling and perform evaluation on a per-token basis. We must point out here although (Benajiba and Rosso, 2008) uses the same ANERcorp corpus that we use, (Benajiba and Rosso, 2008) does not, and we do not have access to the testing data used by them. As a result, this is not an exact comparison. However we wish to obtain a larger picture of the effectiveness of the approach used by SYNERGY, and this comparison serves that purpose reasonably.

System	Prec	Recl	F1
SYNERGY BF w/o PP	0.680	0.502	0.578
SYNERGY BF with PP	0.848	0.600	0.703
SYNERGY GIZA++ w/o PP	0.530	0.702	0.603
SYNERGY GIZA++ with PP	0.761	0.817	0.788
(Benajiba and Rosso, 2008)	0.869	0.727	0.792
(Benajiba et al., 2008)	N/A	N/A	0.835

Table 3: SYNERGY Results for Arabic NER and comparison with other systems

We find that the F1 score achieved by the "GIZA++ with Post-Processing" version of SYNERGY comes quite close to the scores achieved by the state-of-the-art systems. In addition, SYNERGY has the advantage of requiring vastly less time and effort to build and adapt to new languages than other systems. It does not require collecting and annotating an NER training corpus for each new language.

There are three possible types of errors that SYNERGY may produce:

- Named Entities that are lost during translation from the source language to English.
- Errors made by the English NER module of SYNERGY
- A correctly recognized English NE word that gets mapped to the wrong word in the source language document during alignment

It would be interesting to analyze the distribution of SYNERGY's errors across each of these categories. However, that would require the presence of NE gold standard data for the English translations of our Swahili and Arabic test sets in addition to native gold standard data, and since there are no other known systems that employ an MT based approach to NER, such data is not currently available. Therefore, we are unable to perform this analysis.

6. Examples

We illustrate the performance of each stage of SYNERGY on sample Swahili and Arabic sentences. The following NE tagged Swahili sentence is taken from a newswire article in the HCS corpus (Here, we use italics to indicate true Named Entities):

"*Lundenga* aliwataja mawakala ambao wameshatuma maombi kuwa ni kutoka mikoa ya *Iringa Dodoma Mbeya Mwanza* na *Ruvuma*."

. Using Google Translate, SYNERGY translates this sentence to the following English one:

"Lundenga mentioned shatuma agents who have a prayer from the regions of Dodoma Iringa Mwanza Mbeya and Ruvuma."

SYNERGY then adds the following NE labels to this sentence (Here, we use italics to indicate NE labels added by our system):

"*Lundenga* mentioned shatuma agents who have a prayer from the regions of *Dodoma Iringa Mwanza Mbeya* and *Ruvuma*."

Finally, after applying GIZA++ alignment and post-processing, the sentence returned as SYNERGY's final output is:

"*Lundenga* aliwataja mawakala ambao wameshatuma maombi kuwa ni kutoka mikoa ya *Iringa Dodoma Mbeya Mwanza* na *Ruvuma*."

Now, consider the following NE tagged Arabic sentence taken from the ANERcorp corpus, shown here after being transliterated according to the Buckwalter encoding (Buckwalter, 2002):

"n$yr AlY h*h AlZAhrp l¿nhA t$kl Aljw Al*y yEml fyh fryq Alr}ys *jwrj bw$ wrAys* wAlsfyr *jwn bwltwn* fy *AlAmm AlmtHdp* , wAl*y symyz Alkvyr mn AlEnASr Alty qd yqdmhA bED AlAwrwbyyn lrdE tmAdy *AlwlAyAt AlmtHdp* fy AlAstvmAr bqrAr AlEdwAn AlAsrA}yly ElY *lbnAn*."

The NE tagged English translation of this sentence produced by SYNERGY is:

"Refer to this phenomenon because it is the atmosphere with a team of President *George W. Bush Rice* and Ambassador *John Bolton* at the *United Nations* which will recognize a lot of elements that might make some *Europeans* to deter the persistence of the *United States* decision to invest in the *Israeli* aggression on *Lebanon*."

Finally, SYNERGY maps the Named Entities in this sentence back to the original Arabic sentence and gives the following output:

"n$yr AlY h*h AlZAhrp l¿nhA t$kl Aljw Al*y yEml fyh fryq Alr}ys *jwrj bw$ wrAys wAlsfyr jwn bwltwn* fy *AlAmm AlmtHdp* , wAl*y symyz Alkvyr mn AlEnASr Alty qd yqdmhA bED AlAwrwbyyn lrdE tmAdy *AlwlAyAt AlmtHdp* fy AlAstvmAr bqrAr AlEdwAn AlAsrA}yly ElY *lbnAn*."

From these examples, we see that although there are many inaccuracies in both the Swahili and Arabic translations, a vast majority of NE words are preserved across translation and successfully recognized by an English NER system. Moreover, performing NER in English helps us to avoid the difficulties inherent in native Swahili and Arabic NER, e.g. ambiguous function words, recognizing clitics, etc. In other words, SYNERGY addresses the problem of NER for the source language by breaking it into three relatively easier problems: Machine Translation to English, English NER and word alignment between English and the source language.

7. Conclusion and Future Work

We achieve best-case NER F1 scores of 0.788 for Arabic and 0.815 for Swahili. The F1 score for Arabic comes quite close to the state-of-the-art achieved by (Benajiba et al., 2008) and (Benajiba and Rosso, 2008), and the F1 score for Swahili is impressive. Moreover, building SYNERGY and adapting it to new languages is much less expensive than creating an NER training corpus from scratch. Of course, someone had to build the translation system first, but these exist for many more languages than do NER systems. We believe ours is the first freely available Named Entity Recognition system for Swahili[1], and we hope it will be a valuable tool for researchers wishing to work with Swahili text. We intend to use SYNERGY to perform NER for various other resource-scarce languages supported by online translators.

One important language technology that is even less widely available than Named Entity Recognition for many languages is Co-Reference Resolution (CRR), and this is a natural problem to approach using SYNERGY. But unlike NER, in the case of CRR, to test SYNERGY's output we would need a parallel English-Swahili corpus. The SAWA corpus (De Pauw et al., 2009) has not been released yet, and we are not aware of any other such parallel corpus. With its release, this need will be addressed. In the future, we plan to augment SYNERGY to perform CRR for Swahili and also other languages.

[1] `http://www.cs.cmu.edu/~encore/`

8. References

Alchemy API™ (2010). [Online]. Available: http://www.alchemyapi.com (accessed March 2010).

Alias-i. LingPipe™ (2010). [Online]. Available: http://alias-i.com/lingpipe (accessed March 2010).

Benajiba, Y. & Rosso, P. (2008). Arabic Named Entity Recognition using Conditional Random Fields. In *Proceedings of Workshop on HLT and NLP within the Arabic World. LREC 2008*.

Benajiba, Y., Diab, M. & Rosso, P. (2008). Arabic Named Entity Recognition using Optimized Feature Sets. In *Proceedings of the 2008 Conference on Empirical Methods in Natural Language Processing*.

Benajiba, Y., Rosso, P. & Ruiz, J. M. B. (2007). ANERsys: An Arabic Named Entity Recognition system based on Maximum Entropy. In *Proceedings of Computational Linguistics and Intelligent Text Processing*, pp. 143–153.

Buckwalter, T. (2002). Arabic Transliteration. [Online]. Available: http://www.qamus.org/transliteration.htm (accessed March 2010).

Borovikov, A., Borovikov, E., Colquitt, B. & Summers K. (2009). The EDEAL Project for Automated Processing of African Languages. In *Proceedings of MT Summit XII*. Ottawa, Canada.

CoNLL. (2002). CoNLL-2002 shared task: Language-independent named entity recognition. [Online]. Available: http://www.cnts.ua.ac.be/conll2002/ner (accessed March 2010).

CoNLL. (2003). CoNLL-2003 shared task: Language-independent named entity recognition. [Online]. Available: http://www.cnts.ua.ac.be/conll2003/ner (accessed March 2010).

De Pauw, G., de Schryver, G. & Wagacha, P. W. (2006). Data-driven part-of-speech tagging of Kiswahili. In *Proceedings of Text, Speech and Dialogue, 9th International Conference.* Berlin, Germany: Springer Verlag, pp. 197–204.

De Pauw, G., Wagacha, P. W. & de Schryver, G. (2009). The SAWA Corpus: a Parallel Corpus English - Swahili. In *Proceedings of the First Workshop on Language Technologies for African Languages (AfLaT 2009), Association for Computational Linguistics*. Athens, Greece, pp. 9–16.

Finkel, J. R., Grenager, T. & Manning, C. (2005). Incorporating Non-local Information into Information Extraction Systems by Gibbs Sampling. In *Proceedings of the 43rd Annual Meeting of the Association for Computational Linguistics (ACL 2005)*.

Google Translate™ (2010). [Online]. Available: http://www.translate.google.com (accessed March 2010).

Helsinki Corpus of Swahili (HCS). (2004). Institute for Asian and African Studies (University of Helsinki) and CSC Scientific Computing Ltd.

Huang, F. (2006). *Multilingual Named Entity Extraction*

And Translation From Text And Speech. Phd Thesis, LTI, Carnegie Mellon University.

Kamusi Project. (2010). [Online]. Available: http://www.kamusiproject.org (accessed March 2010).

Maamouri, M., Bies, A., Buckwalter, T., Jin, H. & Mekki, W. (2005). Arabic Treebank: Part 3 (full corpus) v 2.0 (MPG + Syntactic Analysis). Linguistic Data Consortium, Philadelphia.

Microsoft Bing Translator™ (2010). [Online]. Available: http://www.microsofttranslator.com (accessed March 2010).

Moses: Open source toolkit for statistical machine translation. [Online]. Available: http://www.statmt.org/moses/ (accessed March 2010).

MUC6. (1995). *Proceedings of the Sixth Message Understanding Conference (MUC6)*. DARPA, Morgan-Kaufmann.

Och, F. J. & Ney, H. (2003). A Systematic Comparison of Various Statistical Alignment Models. *Computational Linguistics*, 29(1), pp. 19–51.

Porter, M. F. (1980). An Algorithm for Suffix Stripping. *Program*, 14(3), pp. 130–137.

Ratinov, L. & Roth, D. (2009). Design Challenges and Misconceptions in Named Entity Recognition. *CoNLL 2009*.

Zhao, B., Bach, N., Lane, I. & Vogel, S. (2007). A Log-linear Block Transliteration Model based on Bi-Stream HMMs. In *Proceedings of NAACL-HLT 2007*. Rochester, NY, USA.