Unsupervised Morphology-Based Vocabulary Expansion

Mohammad Sadegh Rasooli, Thomas Lippincott, Nizar Habash and **Owen Rambow**
Center for Computational Learning Systems
Columbia University, New York, NY, USA
{rasooli,tom,habash,rambow}@ccls.columbia.edu

Abstract

We present a novel way of generating unseen words, which is useful for certain applications such as automatic speech recognition or optical character recognition in low-resource languages. We test our vocabulary generator on seven low-resource languages by measuring the decrease in out-of-vocabulary word rate on a held-out test set. The languages we study have very different morphological properties; we show how our results differ depending on the morphological complexity of the language. In our best result (on Assamese), our approach can predict 29% of the token-based out-of-vocabulary with a small amount of unlabeled training data.

1 Introduction

In many applications in human language technologies (HLT), the goal is to generate text in a target language, using its standard orthography. Typical examples include automatic speech recognition (ASR, also known as STT or speech-to-text), optical character recognition (OCR), or machine translation (MT) into a target language. We will call such HLT applications "target-language generation technologies" (TLGT). The best-performing systems for these applications today rely on training on large amounts of data: in the case of ASR, the data is aligned audio and transcription, plus large unannotated data for the language modeling; in the case of OCR, it is transcribed optical data; in the case of MT, it is aligned bitexts. More data provides for better results. For languages with rich resources, such as English, more data is often the best solution, since the required data is readily available (including bitexts), and the cost of annotating (e.g., transcribing) data is outweighed by the potential significance of the systems that the data will enable. Thus, in HLT, improvements in quality are often brought about by using larger data sets (Banko and Brill, 2001).

When we move to low-resource languages, the solution of simply using more data becomes less appealing. Unannotated data is less readily available: for example, at the time of publishing this paper, 55% of all websites are in English, the top 10 languages collectively account for 90% of web presence, and the top 36 languages have a web presence that covers at least 0.1% of web sites.[1] All other languages (and all languages considered in this paper except Persian) have a web presence of less than 0.1%. Considering Wikipedia, another resource often used in HLT, English has 4.4 million articles, while only 48 other languages have more than 100,000.[2] As attention turns to developing HLT for more languages, including low-resource languages, alternatives to "more-data" approaches become important.

At the same time, it is often not possible to use knowledge-rich approaches. For low-resource languages, resources such as morphological analyzers are not usually available, and even good scholarly descriptions of the morphology (from which a tool could be built) are often not available. The challenge is therefore to use data, but to make do with a small amount of data, and thus to use data better. This paper is a contribution to this goal. Specifically, we address TLGTs, i.e., the types of HLT mentioned above that generate target language text. We propose a new approach to generating unseen words of the target language which have not been seen in the training data. Our approach is entirely unsupervised. It assumes that word-units are specified, typically by whitespace and punctuation.

[1] http://en.wikipedia.org/wiki/Languages_used_on_the_Internet

[2] http://meta.wikimedia.org/wiki/List_of_Wikipedias

Proceedings of the 52nd Annual Meeting of the Association for Computational Linguistics, pages 1349–1359,
Baltimore, Maryland, USA, June 23-25 2014.

Expanding the vocabulary of the target language can be useful for TLGTs in different ways. For ASR and OCR, which can compose words from smaller units (phones or graphically recognized letters), an expanded target language vocabulary can be directly exploited without the need for changing the technology at all: the new words need to be inserted into the relevant resources (lexicon, language model) etc, with appropriately estimated probabilities. In the case of MT into morphologically rich low-resource languages, morphological segmentation is typically used in developing the translation models to reduce sparsity, but this does not guarantee against generating wrong word combinations. The expanded word combinations can be used to extend the language models used for MT to bias against incoherent hypothesized new sequences of segmented words.

Our approach relies on unsupervised morphological segmentation. We do not in this paper contribute to research in unsupervised morphological segmentation; we only use it. The contribution of this paper lies in proposing how to use the results of unsupervised morphological segmentation in order to generate unseen words of the language. We investigate several ways of doing so, and we test them on seven low-resource languages. These languages have very different morphological properties, and we show how our results differ depending on the morphological complexity of the language. In our best result (on Assamese), we show that our approach can predict 29% of the token-based out-of-vocabulary with a small amount of unlabeled training data.

The paper is structured as follows. We first discuss related work in Section 2. We then present our method in Section 3, and present experimental results in Section 4. We conclude with a discussion of future work in Section 5.

2 Related Work

Approaches to Morphological Modeling Computational morphology is a very active area of research with a multitude of approaches that vary in the degree of manual annotation needed, and the amount of machine learning used. At one extreme, we find systems that are painstakingly and carefully designed by hand (Koskenniemi, 1983; Buckwalter, 2004; Habash and Rambow, 2006; Détrez and Ranta, 2012). Next on the continuum, we find work that focuses on defining morphological models with limited lexica that are then extended using raw text (Clément et al., 2004; Forsberg et al., 2006). In the middle of this continuum, we find efforts to learn complete paradigms using fully supervised methods relying on completely annotated data points with rich morphological information (Durrett and DeNero, 2013; Eskander et al., 2013). Next, there is work on minimally supervised methods that use available resources such as dictionaries, bitexts, and other additional morphological annotations (Yarowsky and Wicentowski, 2000; Cucerzan and Yarowsky, 2002; Neuvel and Fulop, 2002; Snyder and Barzilay, 2008). At the other extreme, we find unsupervised methods that learn morphology models from unannotated data (Creutz and Lagus, 2007; Monson et al., 2008; Dreyer and Eisner, 2011; Sirts and Goldwater, 2013).

The work we present in this paper makes no use of any morphological annotations whatsoever, yet we are quite distinct from the approaches cited above. We compare our work to two efforts specifically. First, consider work in automatic morphological segmentation learning from unannotated data (Creutz and Lagus, 2007; Monson et al., 2008). Unlike these approaches which provide segmentations for training data and produce models that can be used to segment unseen words, our approach can generate words that have not been seen in the training data. The focus of efforts is rather complementary: we actually use an off-the-shelf unsupervised segmentation system (Creutz and Lagus, 2007) as part of our approach. Second, consider paradigm completion methods such as the work of Dreyer and Eisner (2011). This effort is closely related to our work although unlike it, we make no assumptions about the data and do not introduce any restrictions along the lines of derivation/inflectional morphology: Dreyer and Eisner (2011) limited their work to verbal paradigms and used annotated training data in addition to basic assumptions about the problem such as the size of the paradigms. In our approach, we have zero annotated information and we do not distinguish between inflectional and derivational morphology, nor do we limit ourselves to a specific part-of-speech (POS).

Vocabulary Expansion in HLT There have been diverse approaches towards dealing with out-of-vocabulary (OOV) words in ASR. In some models, the approach is to expand the lexicon by

adding new words or pronunciations. Ohtsuki et al. (2005) propose a two-run model where in the first run, the input speech is recognized by the reference vocabulary and relevant words are extracted from the vocabulary database and added thereafter to the reference vocabulary to build an expanded lexicon. Word recognition is done in the second run based on the lexicon. Lei et al. (2009) expanded the pronunciation lexicon via generating all possible pronunciations for a word before lattice generation and indexation. There are also other methods for generating abbreviations in voice search systems such as Yang et al. (2012). While all of these approaches involve lexicon expansion, they do not employ any morphological information.

In the context of MT, several researchers have addressed the problem of OOV words by relating them to known in-vocabulary (INV) words. Yang and Kirchhoff (2006) anticipated OOV words that are potentially morphologically related using phrase-based backoff models. Habash (2008) considered different techniques for vocabulary expansion online. One of their techniques learned models of morphological mapping between morphologically rich source words in Arabic that produce the same English translation. This was used to relate an OOV word to a morphologically related INV word. Another technique expanded the MT phrase tables with possible transliterations and spelling alternatives.

3 Morphology-based Vocabulary Expansion

3.1 Approach

Our approach to morphology-based vocabulary expansion consists of three steps (Figure 1). We start with a "training" corpus of (unannotated) words and generate a list of new (unseen) words that expands the vocabulary of the training corpus.

1. **Unsupervised Morphology Segmentation** The first step is to segment each word in the training corpus into sequences of prefixes, stem and suffixes, where the prefixes and suffixes are optional.[3]

2. **FST-based Morphology Expansion** We then construct new word models using the segmented stems and affixes. We explore two different techniques for morphology-based vocabulary expansion that we discuss below. The output of these models is represented as a weighted finite state machine (WFST).

3. **Reranking Models** Given that the size of the expanded vocabulary can be quite large and it may include a lot of over-generation, we rerank the expanded set of words before taking the top n words to use in downstream processes. We consider four reranking conditions which we describe below.

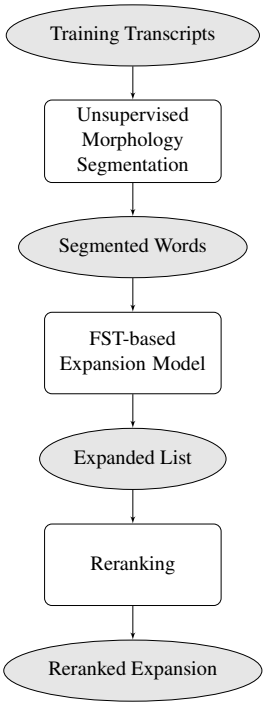


Figure 1: The flowchart of the lexicon expansion system.

3.2 Morphology Expansion Techniques

As stated above, the input to the morphology expansion step is a list of words segmented into morphemes: zero or more prefixes, one stem, and zero or more suffixes. Figure 2a presents an example of such input using English words (for clarity).

We use two different models of morphology expansion in this paper: Fixed Affix model and Bigram Affix model.

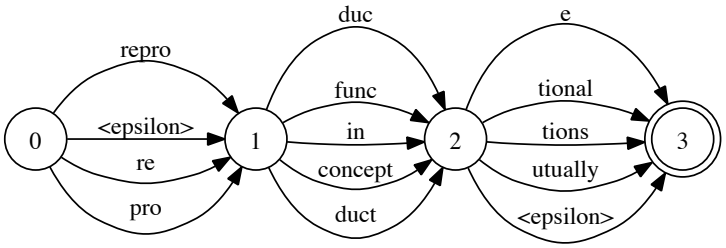
3.2.1 Fixed Affix Expansion Model

In the Fixed Affix model, we construct a set of fused prefixes from all the unique prefix sequences in the training data; and we similarly construct a

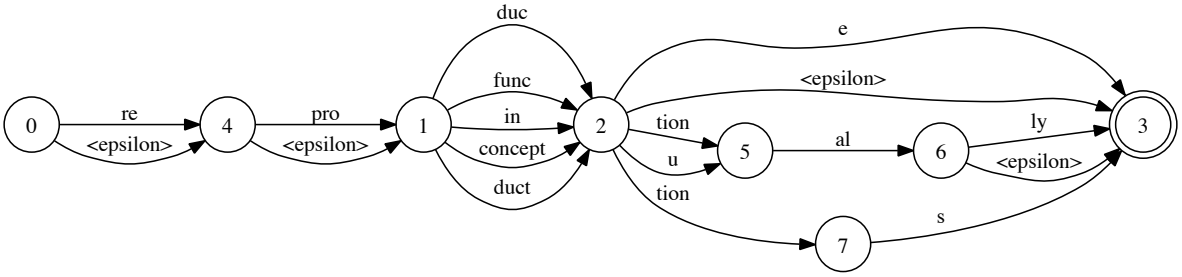
[3] In this paper, we use an off-the-shelf system for this step but plan to explore new methods in the future, such as joint segmentation and expansion.

re+ pro+ duc +e
func +tion +al
re+ duc +e
re+ duc +tion +s
in
pro+ duct
concept +u +al + ly

(a) Training data with morpheme boundaries. Prefixes end with and suffixes start with "+" signs.



(b) FST for the Fixed Affix expansion model



(c) FST for the Bigram Affix expansion model

Figure 2: Two models of word generation from morphologically annotated data. In our experiments, we used weighted finite state machine. We use character-based WFST in the implementation to facilitate analyzing inputs as well as word generation.

set of fused suffixes from all the unique suffix sequences in the training data. In other words, we simply pick characters from beginning of the word up to the first stem as the prefix and characters from the first suffix to the end of the word as the suffix. Everything in the middle is the stem. In this model, each word has one single prefix and one single suffix (each of which can be empty independently). The Fixed Affix model is simply the concatenation of the disjunction of all prefixes with the disjunction of all stems and the disjunction of all suffixes into one FST:

$$\textit{prefix} \rightarrow \textit{stem} \rightarrow \textit{suffix}$$

The morpheme paths in the FST are weighted to reflect their probability in the training corpus.[4] Figure 2b exemplifies a Fixed Affix model derived from the example training data in Figure 2a.

[4]We convert the probability into a cost by taking the negative of the log of the probability.

3.2.2 Bigram Affix Expansion Model

In the Bigram Affix model, we do the same for the stem as in the Fixed Affix model, but for prefixes and suffixes, we create a bigram language model in the finite state machine. The advantage of this technique is that unseen compound affixes can be generated by our model. For example, the Fixed Affix model in Figure 2b cannot generate the word *func+tion+al+ly* since the suffix *+tionally* is not seen in the training data. However, this word can be generated in the Bigram Affix model as shown in Figure 2c: there is a path passing $0 \rightarrow 4 \rightarrow 1 \rightarrow 2 \rightarrow 5 \rightarrow 6 \rightarrow 3$ in the FST that can produce this word. We expect this model to have better recall for generating new words in the language because of its affixation flexibility.

3.3 Reranking Techniques

The expanded models allow for a large number of words to be generated. We limit the number of vocabulary expansion using different thresholds after reranking or reweighing the WFSTs generated

above. We consider four reranking conditions.

3.3.1 No Reranking (NRR)

The baseline reranking option is no reranking (NRR). In this approach we use the weights in the WFST, which are based on the independent prefix/stem/suffix probabilities, to determine the ranking of the expanded vocabulary.

3.3.2 Trigraph-based Reweighting (W∘Tr)

We reweight the weights in the WFST model (Fixed or Bigram) by composing it with a *letter trigraph language model* ($W \circ \text{Tr}$). A letter trigraph LM is itself a WFST where each trigraph (a sequence of three consequent letters) has an associated weight equal to its negative log-likelihood in the training data. This reweighting allows us to model preferences of sequences of word letters seen more in the training data. For example, in a word like *producttions*, the trigraphs *ctt* and *tti* are very rare and thus decrease its probability.

3.3.3 Trigraph-based Reranking (TRR)

When we compose our initial WFST with the trigraph FST, the probability of each generated word from the new FST is equal to the product of the probability of its morphemes and the probabilities of each trigraph in that word. This basically makes the model prefer shorter words and may degrade the effect of morphology information. Instead of reweighting the WFST, we get the n-best list of generated words and rerank them using their trigraph probabilities. We will refer to this technique as TRR.

3.3.4 Reranking Morpheme Boundaries (BRR)

The last reranking technique reranks the n-best generated word list with trigraphs that are incident on the morpheme boundaries (in case of Bigram Affix model, the last prefix and first suffix). The intuition is that we already know that any morpheme that is generated from the morphology FST is already seen in the training data but the boundary for different morphemes are not guaranteed to be seen in the training data. For example, for the word *producttions*, we only take into account the trigraphs *rod*, *odu*, *ctt* and *tti* instead of all possible trigraphs. We will refer to this technique as BRR.

4 Evaluation

4.1 Evaluation Data and Tools

Evaluation Data The IARPA Babel program is a research program for developing rapid spoken detection systems for under-resourced languages (Harper, 2013). We use the IARPA Babel program limited language pack data which consists of 20 hours of telephone speech with transcription. We use six languages which are known to have rich morphology: Assamese (IARPA-babel102b-v0.5a), Bengali (IARPA-babel103b-v0.4b), Pashto (IARPA-babel104b-v0.4bY), Tagalog (IARPA-babel106-v0.2g), Turkish (IARPA-babel105b-v0.4) and Zulu (IARPA-babel206b-v0.1e). Speech annotation such as silences and hesitations are removed from transcription and all words are turned into lower-case (for languages using the Roman script – Tagalog, Turkish and Zulu). Moreover, in order to be able to perform a manual error analysis, we include a language that has rich morphology and of which the first author is a native speaker: Persian. We sampled data from the training and development set of the Persian dependency treebank (Rasooli et al., 2013) to create a comparable seventh dataset in Persian. Statistics about the datasets are shown in Table 1. We also conduct further experiments on just verbs and nouns in the data set for Persian (Persian-N and Persian V). As shown in Table 1, the training data is very small and the OOV rate is high especially in terms of types. For some languages that have richer morphology such as Turkish and Zulu, the OOV rate is much higher than other languages.

Word Generation Tools and Settings For unsupervised learning of morphology, we use Morfessor CAT-MAP (v. 0.9.2) which was shown to be a very accurate morphological analyzer for morphologically rich languages (Creutz and Lagus, 2007). In order to be able to analyze Unicode-based data, we convert each character in our dataset to some conventional ASCII character and then train Morfessor on the mapped dataset; after finishing the training, we map the data back to the original character set. We use the default setting in Morfessor for unsupervised learning.

For preparing the WFST, we use OpenFST (Riley et al., 2009). We get the top one million shortest paths (i.e., least costly paths of words) and apply our reranking models on them. It is worth pointing out that our WFSTs are character-based

Language	Training Data		Development Data			
	Type	Token	Type	Token	Type OOV%	Token OOV%
Assamese	8694	73151	7253	66184	49.57	8.28
Bengali	9460	81476	7794	70633	50.65	8.47
Pashto	6968	115069	6135	108137	44.89	4.25
Persian	14047	71527	10479	42939	44.16	12.78
Tagalog	6213	69577	5480	64334	54.95	7.81
Turkish	11985	77128	9852	67042	56.84	12.34
Zulu	15868	65655	13756	57141	68.72	21.76
Persian-N	9204	31369	7502	18816	46.36	22.11
Persian-V	2653	11409	1332	7318	41.07	9.01

Table 1: Statistics of training and development data for morphology-based unsupervised word generation experiments.

and thus we also have a morphological analyzer that can give all possible segmentations for a given word. By running the morphological analyzer on the OOVs, we can have the potential upper bound of OOV reduction by the system (labeled "∞" in Tables 2 and 3).

4.2 Lexicon Expansion Results

The results for lexicon expansion are shown in Table 2 for types and Table 3 for tokens.

We use the trigraph WFST as our baseline model. This model does not use any morphological information. In this case, words are generated according to the likelihood of their trigraphs, without using any information from the morphological segmentation. We call this model the trigraph WFST (Tr. WFST). We consistently have better numbers than this baseline in all of our models except for Pashto when measured by tokens. ∞ is the upper-bound OOV reduction for our expansion model: for each word in the development set, we ask if our model, without any vocabulary size restriction at all, could generate it.

The best results (again, except for Pashto) are achieved using one of the three reranking methods (reranking by trigraph probabilities or morpheme boundaries) as opposed to doing no reranking. To our surprise, the Fixed Affix model does a slightly better job in reducing out of vocabulary than the Bigram Affix model. We can also see from the results that reranking in general is very effective.

We also compare our models with the case that there is much more training data and we do not do vocabulary expansion at all. In Table 2 and Table 3, "FP" indicates the full language pack for the Babel project data which is approximately six to eight times larger than the limited pack training data, and the full training data for Persian which is approximately five times larger. We see that the larger training data outperforms our methods in all languages. However, from the results of ∞, which is the upper-bound OOV reduction by our expansion model, for some languages such as Assamese, our numbers are close to the FP results and for Zulu it is even better than FP.

We also study how OOV reduction is affected by the size of the generated vocabulary. The trends for different sizes of the lexicon expansion by Fixed Affix model that is reranked by trigraph probabilities is shown in Figure 3. As seen in the results, for languages that have richer morphology, it is harder to achieve results near to the upper bound. As an outlier, morphology does not help for Pashto. One possible reason might be that based on the results in Table 4, Morfessor does not explore morphology in Pashto as well as other languages.

Morphological Complexity As for further analysis, we can study the correlation between morphological complexity and hardness of reducing OOVs. Much work has been done in linguistics to classify languages (Sapir, 1921; Greenberg, 1960). The common wisdom is that languages are not either agglutinative or fusional, but are on a spectrum; however, no work to our knowledge places all languages (or at least the ones we worked on) on such a spectrum. We propose several metrics. First, we can consider the number of unique affixival morphemes in each language, as determined by Morfessor. As shown in Table 4 ($|pr| + |sf|$), Zulu has the most morphemes and Pashto the fewest. A second possible metric of the

Language	Tr. WFST	Fixed Affix Model					Bigram Affix Model					FP
		NRR	W∘Tr	TRR	BRR	∞	NRR	W∘Tr	TRR	BRR	∞	
Assamese	15.94	24.03	**28.46**	28.15	27.15	48.07	23.50	28.15	27.84	26.59	51.02	50.96
Bengali	15.68	20.09	**24.75**	24.49	22.54	40.98	21.78	24.65	24.67	23.51	42.55	48.83
Pashto	18.70	19.03	19.28	19.24	18.63	25.13	**19.43**	18.81	18.92	18.77	25.24	64.96
Persian	12.83	18.95	18.39	19.30	**19.99**	50.11	18.58	18.09	18.65	18.84	53.13	58.45
Tagalog	11.39	14.61	16.51	16.21	**16.81**	35.64	14.45	16.01	15.81	16.74	38.72	53.64
Turkish	07.75	09.11	**14.79**	**14.79**	14.71	55.48	09.04	13.63	14.34	13.52	66.54	53.54
Zulu	07.63	11.87	12.96	**13.87**	13.68	66.73	12.04	12.35	13.69	13.75	82.38	35.62
Average	12.85	16.81	**19.31**	**19.31**	19.07	46.02	17.02	18.81	19.13	18.81	51.37	52.29
Persian-N	14.86	**24.67**	22.74	22.83	24.15	37.32	23.78	21.68	22.51	23.32	38.38	-
Persian-V	54.84	68.19	72.39	**73.49**	71.12	80.44	67.28	71.48	72.58	70.02	80.62	-

Table 2: Type-based expansion results for the 50k-best list for different models. Tr. WFST stands for trigraph WFST, NRR for no reranking, W∘Tr for trigraph reweighting, TRR for trigraph-based rereanking, BRR for reranking morpheme boundary, and ∞ for the upper bound of OOV reduction via lexicon expansion if we produce all words. FP (full-pack data) shows the effect of using bigger data with the size of about seven times larger than our data set, instead of using our unsupervised approach.

Language	Tr. WFST	Fixed Affix Model					Bigram Affix Model					FP
		NRR	W∘Tr	TRR	BRR	∞	NRR	W∘Tr	TRR	BRR	∞	
Assamese	18.07	25.70	**29.43**	29.12	28.13	47.88	25.34	29.06	28.82	27.64	50.31	58.03
Bengali	17.79	20.91	**25.61**	25.27	23.65	40.60	22.58	25.20	25.41	24.77	42.22	55.92
Pashto	**21.27**	19.40	19.94	19.92	18.59	25.45	19.68	19.40	19.29	18.72	25.58	71.46
Persian	14.78	20.77	20.32	21.30	**22.03**	51.00	20.63	19.72	20.61	20.95	54.01	63.10
Tagalog	12.88	14.55	**16.88**	16.36	16.60	33.95	14.37	16.12	16.12	16.38	37.07	61.53
Turkish	09.97	11.42	**17.82**	17.67	17.23	56.54	11.05	16.82	17.41	15.98	66.54	59.68
Zulu	08.85	13.70	14.72	15.62	**15.67**	68.07	13.70	14.07	15.47	15.60	87.90	41.27
Average	14.80	18.06	20.67	**20.75**	20.27	44.78	18.19	20.48	20.45	20.01	51.95	58.71
Persian-N	16.82	**26.46**	24.42	24.56	25.71	38.40	25.69	23.50	24.20	25.04	39.41	–
Persian-V	60.09	71.47	75.57	**76.48**	73.60	82.55	70.56	74.81	75.72	72.53	82.70	–

Table 3: Token-based expansion results for the 50k-best list for different models. Abbreviations are the same as Table 2.

complexity of the morphology is by calculating the average number of unique prefix-suffix pairs in the training data after morpheme segmentation which is shown as $|If|$ in Table 4. Finally, a third possible metric is the number of all possible words that can be generated ($|L|$). These three metrics correlate fairly well across the languages.

The metrics we propose also correlate with commonly accepted classifications: e.g., Zulu and Turkish (highly agglutinative) have higher scores in terms of our $|pr| + |sf|$, $|If|$ and $|L|$ metrics in Table 4 than other languages. The results from full language packs in Table 3 also show that there is a reverse interaction of morphological complexity and the effect of blindly adding more data. Thus for morphologically rich languages, adding more data is less effective than for languages with poor morphology.

The size of the languages ($|L|$) suggests that we are suffering from vast overgeneration; we overgenerate because in our model any affix can attach to any stem, which is not in general true. Thus there is a lack of linguistic knowledge such as paradigm information (Stump, 2001) for each word category in our model. In other words, all morphemes are treated the same in our model which is not true in natural languages. One way to tackle this problem is through an unsupervised POS tagger. The challenge here is that fully unsupervised POS taggers (without any tag dictionary) are not very accurate (Christodoulopoulos et al., 2010). Another way is through using joint mor-

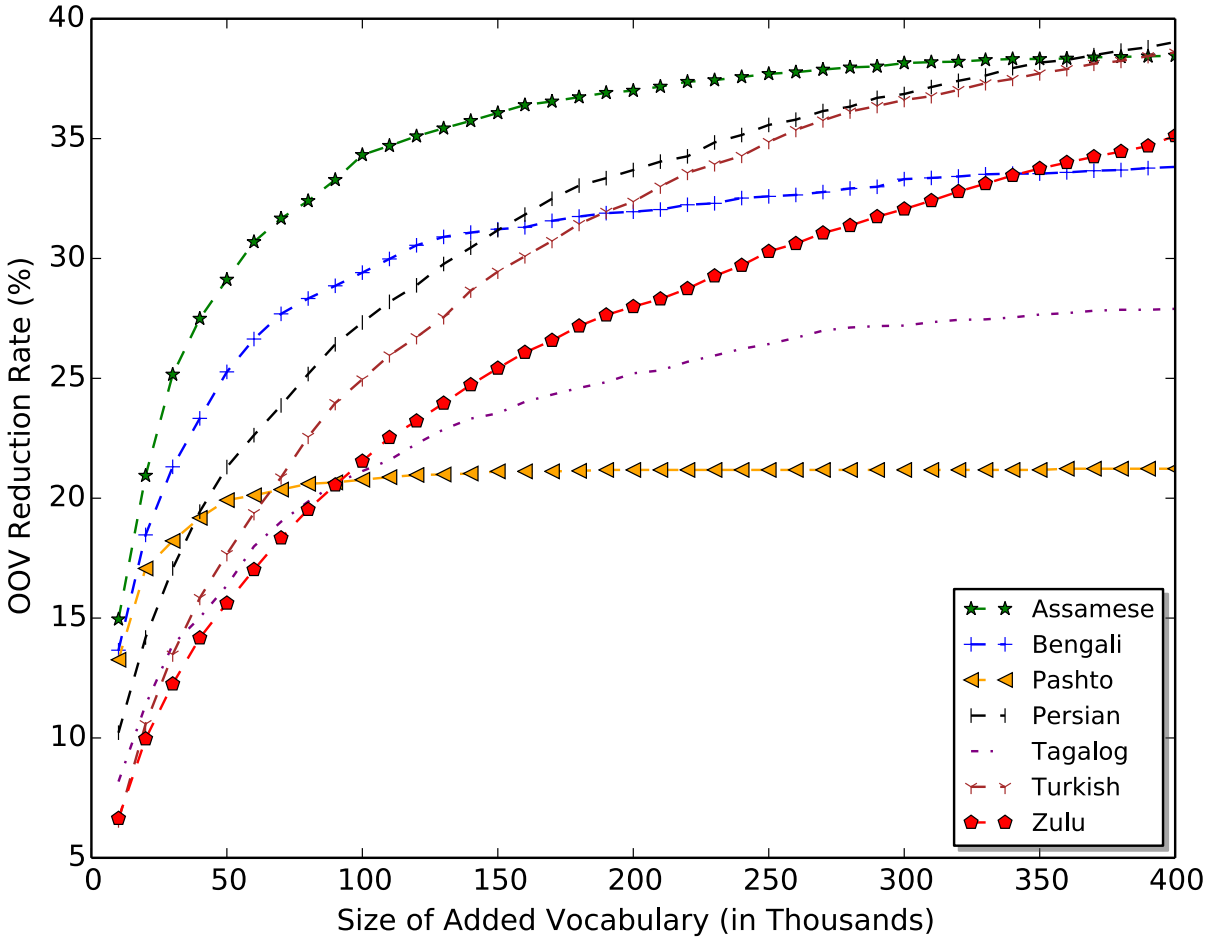


Figure 3: Trends for token-based OOV reduction with different sizes for the Fixed Affix model with trigraph reranking.

Language	$\|pr\|$	$\|stm\|$	$\|sf\|$	$\|L\|$	$\|If\|$
Assamese	4	4791	564	10.8M	1.8
Bengali	3	6496	378	7.4M	1.5
Pashto	1	5395	271	1.5M	1.3
Persian	49	6998	538	184M	2.0
Tagalog	179	4259	299	228M	1.5
Turkish	45	5266	1801	427M	2.3
Zulu	2254	5680	427	5.5B	2.8
Persian-N	3	6121	268	4.9M	1.5
Persian-V	43	788	44	1.5M	3.4

Table 4: Information about the number of unique morphemes in the Fixed Affix model for each dataset including empty affixes. $|L|$ shows the upper bound of the number of possible unique words that can be generated from the word generation model. $|If|$ is the average number of unique prefix-suffix pairs (including empty pairs) for each stem.

phology and tagging models such as Frank et al. (2013).

Error Analysis on Turkish Unfortunately for most languages we could not find an available rule-based or supervised morphological analyzer to verify the words generated by our model. The only available tool for us is a Turkish finite-state morphological analyzer (Oflazer, 1996) implemented with the Xerox FST toolkit (Beesley and Karttunen, 2003). As we can see in Table 5, the system with the largest proportion of correct generated words reranks the expansion with trigraph probabilities using a Fixed Affix model. Results also show that we are overgenerating many nonsense words that we ought to be pruning from our results. Another observation is that the recognition percentage of the morphological analyzer on INV words is much higher than on OOVs, which shows that OOVs in Turkish dataset are much harder to analyze.

Model		Precision
Tr. WFST		17.19
Fixed Affix Model	NRR	13.36
	W∘Tr	25.66
	TRR	**26.30**
	BRR	25.14
Bigram Affix Model	NRR	12.94
	W∘Tr	24.21
	TRR	25.39
	BRR	23.45
Development	words	89.30
	INVs	95.44
	OOVs	84.64

Table 5: Results from running a hand-crafted Turkish morphological analyzer (Oflazer, 1996) on different expansions and on the development set. Precision refers to the percentage of the words are recognized by the analyzer. The results on development are also separated into INV and OOV.

Error Analysis on Persian From the best 50k word result for Persian (Fixed Affix Model:BRR), we randomly picked 200 words and manually analyzed them. 89 words are correct (45.5%) where 55.0% of these words are from noun affixation, 23.6% from verb clitics, 9.0% from verb inflections, 5.6% from incorrect affixations that accidentally resulted in possible words, 4.5% from uninflected stems, and a few from adjective affixation. Among incorrectly generated words, 65.8% are from combining a stem of one POS with affixes from another POS (e.g., attaching a noun affix to a verb stem), 14.4% from combining a stem with affixes which are compatible with POS but not allowed for that particular stem (e.g., there is a noun suffix that can only attach to a subset of noun stems), 9.0% are from wrong affixes produced by Morfessor and others are from incorrect vowel harmony or double affixation.

In order to study the effect of vocabulary expansion more deeply, we trained a subset of all nouns and verbs in the same dataset (also shown in Table 1). Verbs in Persian have rich but more or less regular morphology, while nouns, which have many irregular cases, have rich morphology but not as rich as verbs. The results in Table 4 show that Morfessor captures these phenomena. Furthermore, our results in Table 2 and Table 3 show that our performance on OOV reduction with verbs is far superior to our performance with nouns. We also randomly picked 200 words from each of the experiments (noun and verbs) to study the degree of correctness of generated forms. For nouns, 94 words are correct and for verbs only 71 words are correct. Most verb errors are due to incorrect morpheme extraction by Morfessor. In contrast, most noun errors result from affixes that are only compatible with a subset of all possible noun stems. This suggests that if we conduct experiments using more accurate unsupervised morphology and also have a more fine-grained paradigm completion model, we might improve our performance.

5 Conclusion and Future Work

We have presented an approach to generating new words. This approach is useful for low-resource, morphologically rich languages. It provides words that can be used in HLT applications that require target-language generation in this language, such as ASR, OCR, and MT. An implementation of our approach, named BabelGUM (Babel General Unsupervised Morphology), will be publicly available. Please contact the authors for more information.

In future work we will explore the possibility of jointly performing unsupervised morphological segmentation with clustering of words into classes with similar morphological behavior. These classes will extend POS classes. We will tune the system for our purposes, namely OOV reduction.

Acknowledgements

We thank Anahita Bhiwandiwalla, Brian Kingsbury, Lidia Mangu, Michael Picheny, Benoît Sagot, Murat Saraclar, and Géraldine Walther for helpful discussions. The project is supported by the Intelligence Advanced Research Projects Activity (IARPA) via Department of Defense U.S. Army Research Laboratory (DoD/ARL) contract number W911NF-12-C-0012. The U.S. Government is authorized to reproduce and distribute reprints for Governmental purposes notwithstanding any copyright annotation thereon. Disclaimer: The views and conclusions contained herein are those of the authors and should not be interpreted as necessarily representing the official policies or endorsements, either expressed or implied, of IARPA, DoD/ARL, or the U.S. Government.

References

Michele Banko and Eric Brill. 2001. Scaling to very very large corpora for natural language disambiguation. In *Proceedings of the 39th Annual Meeting on Association for Computational Linguistics*, ACL '01, pages 26–33, Stroudsburg, PA, USA. Association for Computational Linguistics.

Kenneth R Beesley and Lauri Karttunen. 2003. Finite-state morphology: Xerox tools and techniques. *CSLI, Stanford*.

Tim Buckwalter. 2004. Buckwalter Arabic Morphological Analyzer Version 2.0. LDC catalog number LDC2004L02, ISBN 1-58563-324-0.

Christos Christodoulopoulos, Sharon Goldwater, and Mark Steedman. 2010. Two decades of unsupervised pos induction: How far have we come? In *Proceedings of the 2010 Conference on Empirical Methods in Natural Language Processing*, pages 575–584. Association for Computational Linguistics.

Lionel Clément, Benoît Sagot, and Bernard Lang. 2004. Morphology based automatic acquisition of large-coverage lexica. In *Proceedings of the Fourth International Conference on Language Resources and Evaluation (LREC'04)*. European Language Resources Association (ELRA).

Mathias Creutz and Krista Lagus. 2007. Unsupervised models for morpheme segmentation and morphology learning. *ACM Transactions on Speech and Language Processing (TSLP)*, 4(1):3.

Silviu Cucerzan and David Yarowsky. 2002. Bootstrapping a multilingual part-of-speech tagger in one person-day. In *The 6th Conference on Natural Language Learning (CoNLL-2002)*, pages 1–7.

Grégoire Détrez and Aarne Ranta. 2012. Smart paradigms and the predictability and complexity of inflectional morphology. In *Proceedings of the 13th Conference of the European Chapter of the Association for Computational Linguistics*, pages 645–653. Association for Computational Linguistics.

Markus Dreyer and Jason Eisner. 2011. Discovering morphological paradigms from plain text using a dirichlet process mixture model. In *Proceedings of the 2011 Conference on Empirical Methods in Natural Language Processing*, pages 616–627. Association for Computational Linguistics.

Greg Durrett and John DeNero. 2013. Supervised learning of complete morphological paradigms. In *Proceedings of the 2013 Conference of the North American Chapter of the Association for Computational Linguistics: Human Language Technologies*, pages 1185–1195. Association for Computational Linguistics.

Ramy Eskander, Nizar Habash, and Owen Rambow. 2013. Automatic extraction of morphological lexicons from morphologically annotated corpora. In *Proceedings of the 2013 Conference on Empirical Methods in Natural Language Processing*, pages 1032–1043, Seattle, Washington, USA, October. Association for Computational Linguistics.

Markus Forsberg, Harald Hammarström, and Aarne Ranta. 2006. Morphological lexicon extraction from raw text data. *Advances in Natural Language Processing*, pages 488–499.

Stella Frank, Frank Keller, and Sharon Goldwater. 2013. Exploring the utility of joint morphological and syntactic learning from child-directed speech. In *Proceedings of the 2013 Conference on Empirical Methods in Natural Language Processing*, pages 30–41. Association for Computational Linguistics.

Joseph H Greenberg. 1960. A quantitative approach to the morphological typology of language. *International journal of American linguistics*, pages 178–194.

Nizar Habash and Owen Rambow. 2006. MAGEAD: A morphological analyzer and generator for the Arabic dialects. In *Proceedings of the 21st International Conference on Computational Linguistics and 44th Annual Meeting of the Association for Computational Linguistics*, pages 681–688, Sydney, Australia.

Nizar Habash. 2008. Four techniques for online handling of out-of-vocabulary words in Arabic-English statistical machine translation. In *Proceedings of the 46th Annual Meeting of the Association for Computational Linguistics on Human Language Technologies: Short Papers*, pages 57–60. Association for Computational Linguistics.

Mary Harper. 2013. The babel program and low resource speech technology. In *Automatic Speech Recognition and Understanding Workshop (ASRU) Invited talk*.

Kimmo Koskenniemi. 1983. Two-Level Model for Morphological Analysis. In *Proceedings of the 8th International Joint Conference on Artificial Intelligence*, pages 683–685.

Xin Lei, Wen Wang, and Andreas Stolcke. 2009. Data-driven lexicon expansion for Mandarin broadcast news and conversation speech recognition. In *International conference on Acoustics, Speech and Signal Processing (ICASSP)*, pages 4329–4332.

Christian Monson, Jaime Carbonell, Alon Lavie, and Lori Levin. 2008. Paramor: Finding paradigms across morphology. *Advances in Multilingual and Multimodal Information Retrieval*, pages 900–907.

Sylvain Neuvel and Sean A Fulop. 2002. Unsupervised learning of morphology without morphemes. In *Proceedings of the ACL-02 workshop on Morphological and phonological learning-Volume 6*, pages 31–40. Association for Computational Linguistics.

Kemal Oflazer. 1996. Error-tolerant finite-state recognition with applications to morphological analysis and spelling correction. *Computational Linguistics*, 22(1):73–89.

Katsutoshi Ohtsuki, Nobuaki Hiroshima, Masahiro Oku, and Akihiro Imamura. 2005. Unsupervised vocabulary expansion for automatic transcription of broadcast news. In *International conference on Acoustics, Speech and Signal Processing (ICASSP)*, pages 1021–1024.

Mohammad Sadegh Rasooli, Manouchehr Kouhestani, and Amirsaeid Moloodi. 2013. Development of a Persian syntactic dependency treebank. In *Proceedings of the 2013 Conference of the North American Chapter of the Association for Computational Linguistics: Human Language Technologies*, pages 306–314. Association for Computational Linguistics.

Michael Riley, Cyril Allauzen, and Martin Jansche. 2009. Openfst: An open-source, weighted finite-state transducer library and its applications to speech and language. In *Human Language Technologies Tutorials: The 2009 Annual Conference of the North American Chapter of the Association for Computational Linguistics*, pages 9–10.

Edward Sapir. 1921. *Language: An introduction to the study of speech*. Harcourt, Brace and company (New York).

Kairit Sirts and Sharon Goldwater. 2013. Minimally-supervised morphological segmentation using adaptor grammars. *Transactions for the ACL*, 1:255–266.

Benjamin Snyder and Regina Barzilay. 2008. Unsupervised multilingual learning for morphological segmentation. In *Proceedings of the 46th annual meeting of the association for computational linguistics: Human language Technologies (ACL-HLT)*, pages 737–745. Association for Computational Linguistics.

Gregory T. Stump. 2001. *A theory of paradigm structure*. Cambridge.

Mei Yang and Katrin Kirchhoff. 2006. Phrase-based backoff models for machine translation of highly inflected languages. In *Proceedings of Conference of the European Chapter of the Association for Computational Linguistics (EACL)*, pages 41–48, Trento, Italy.

Dong Yang, Yi-Cheng Pan, and Sadaoki Furui. 2012. Vocabulary expansion through automatic abbreviation generation for Chinese voice search. *Computer Speech & Language*, 26(5):321–335.

David Yarowsky and Richard Wicentowski. 2000. Minimally supervised morphological analysis by multimodal alignment. In *Proceedings of the 38th Annual Meeting on Association for Computational Linguistics*, pages 207–216.