Modeling Morpheme Triplets with a Three-level Hierarchical Dirichlet Process

Serkan Kumyol
Cognitive Science Department, Informatics Inst.
Middle East Technical University (ODTÜ)
06800, Ankara, Turkey
Email: s.kumyols@gmail.com

Burcu Can
Department of Computer Engineering
Hacettepe University, Beytepe
06800, ANKARA, Turkey
Email: burcucan@cs.hacettepe.edu.tr

***Abstract*—Morphemes are not independent units and attached to each other based on morphotactics. However, they are assumed to be independent from each other to cope with the complexity in most of the models in the literature. We introduce a language independent model for unsupervised morphological segmentation using hierarchical Dirichlet process (HDP). We model the morpheme dependencies in terms of morpheme trigrams in each word. Trigrams, bigrams and unigrams are modeled within a three-level HDP, where the trigram Dirichlet process (DP) uses the bigram DP and bigram DP uses unigram DP as the base distribution. The results show that modeling morpheme dependencies improve the F-measure noticeably in English, Turkish and Finnish.**

***Index Terms*—morphological segmentation; unsupervised learning; non-parametric Bayesian methods; Dirichlet process**

I. Introduction

Morphological segmentation is the task of splitting words into their smallest meaning bearing units. For example, the word *bookings* is split into *book*, *ing*, and *s*, which are called morphemes.

Morphological segmentation is vital in many natural language processing (NLP) tasks, such as machine translation, information retrieval or question answering. It becomes unfeasible to model word forms in NLP tasks due to sparsity especially in agglutinating languages. Having many word forms brings out-of-vocabulary (OOV) issue in those languages severely. Therefore, the usual procedure is to mitigate sparsity by obtaining morphemes of words via morphological segmentation and modeling morphemes instead of words.

Morphological segmentation is treated both as a supervised or unsupervised learning problem in the literature. Many linguistic features (orthographic, syntactic, semantic etc.) have been used in learning morphemes. Regardless of the features or the learning scheme used, morphemes have been usually considered independent in most of the models. However, morphology has its own grammar rules, which are called morphotactics. Therefore, the transition between morphemes is a significant linguistic feature which has been underused in morphological segmentation although it plays an important role in language processing.

In this paper, we model morpheme transitions as a morpheme trigram language model by adopting a non-parametric model, i.e. Dirichlet processes. The results show that adopting higher morpheme n-grams leads to a better accuracy in morphological segmentation task, which shows that morphemes have got a long distance dependency (i.e. three in this paper).

The paper is organized as follows: Section II addresses the related work in the literature, section III describes the mathematical model based on HDP, section IV explains the inference of the model, section V presents the experiments and results with a comparison to other systems, and finally section VI concludes the paper with a brief discussion and potential future work.

II. Related Work

Since unsupervised morphological segmentation does not need any annotated (i.e. morphologically analyzed) data, it has been applied using different learning schemes in the literature. Creutz and Lagus [1] propose the well-known unsupervised morphological segmentation system called *Morfessor*, which is based on minimum description length and forms the baseline of the *Morfessor* family. *Morfessor* has several versions based on different learning mechanisms (a maximum likelihood (ML) based version [2], a maximum a posteriori (MAP) based version [3] and a version based on capturing allomorphs [4]).

Goldwater et al. [5] develop a two-stage model where the types (i.e. morphemes) are created by a generator and the frequency of the types are modified by an adaptor in order to generate a power-law distribution using a Pitman-Yor process.

Can and Manandhar [6] propose a hierarchical Dirichlet process model to capture morphological paradigms that are structured within a tree hierarchy. Each node on the tree denotes a morphological paradigm that consists of a stem and a suffix list.

Narasimhan et al. [7] develop a log linear model by modeling the words through parent-child relations that have a semantic relation. The semantic relation gives a clue on the word that are derived from each other. For example, *booking* is the parent of *book*. The semantic similarity between the word forms are computed using neural word embeddings obtained from an open source tool called word2vec [8] in their model.

Snyder and Barzilay [9] introduce a non-parametric based approach that learns morphological segmentation through a

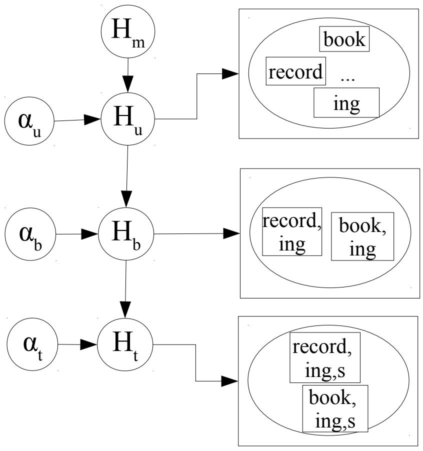

Fig. 1. The plate diagram of the three-level HDP. Unigrams are generated from H_u, bigrams are generated from H_b and trigrams are generated from H_t.

multilingual corpus by capturing aligned morphemes between two different languages.

Most of the models in the literature assume that morphemes are independent from each other. Creutz and Lagus [3] model the transitions between morpheme types (i.e. stem, suffix, or prefix). However, the transitions between morphemes are not modeled in their system. Can et al. [10] introduce a bigram model recently that shows binary transitions reduce sparsity.

Aksan et al. [11] address the multi-morpheme frequencies in Turkish. According to their statistical findings in a Turkish corpus, multi-morpheme (trigrams, fourgrams, and even more) sequences in agglutinating languages have a high frequency and this can be used in morphological segmentation. Here we adopt this information in order to model morpheme transitions in terms of morpheme trigrams by going a bit further by using a three-level hierarchical Dirichlet process.

III. Model Definition

In our trigram model, we assume that each morpheme is dependent on the previous two morphemes as follows:

$$\begin{aligned} p(w &= m_0 + m_1 + m_2 + \cdots + m_z) \\ &= p(m_0)p(m_1|m_0)\prod_{i=2}^{z} p(m_i|m_{i-1}, m_{i-2}) \end{aligned} \quad (1)$$

where m denotes the morphemes (possibly m_0 being the stem) that belong to word w. We generate unigrams, bigrams and trigrams through different DPs. The full mathematical model is defined as given below:

$$\begin{aligned} H_u &\sim DP(\alpha_u, H_m) \\ m_i &\sim H_u \\ H_b &\sim DP(\alpha_b, H_u) \\ m_i|\ m_{i-1} &\sim H_b \\ H_t &\sim DP(\alpha_t, H_b) \\ m_i|\ m_{i-1}, m_{i-2} &\sim H_t \end{aligned} \quad (2)$$

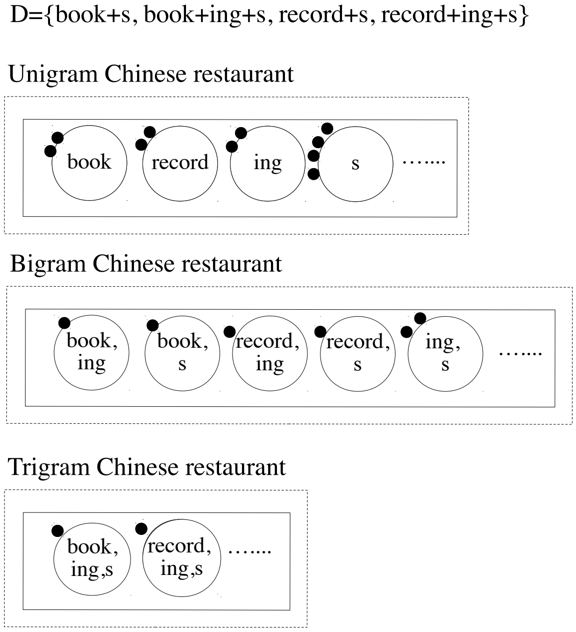
D={book+s, book+ing+s, record+s, record+ing+s}
Unigram Chinese restaurant
book
record
ing
s
....
Bigram Chinese restaurant
book, ing
book, s
record, ing
record, s
ing, s
....
Trigram Chinese restaurant
book, ing,s
record, ing,s
....

Fig. 2. An example of unigram, bigram and trigram Chinese restaurants that consist of tables serving morpheme unigrams, bigrams and trigrams respectively.

where H_u, H_b and H_t correspond to the distributions drawn from the Dirichlet processes that generate unigrams, bigrams, and trigrams respectively The plate diagram of the full model is given in Figure 1. H_m is the base distribution of the first level Dirichlet process and it is in the form of a geometric distribution based on the morpheme length with a parameter γ:

$$H_m(m_i) = \gamma^{|m_i|} \quad (3)$$

where $|m_i|$ is the number of letters in m_i.

From the Chinese restaurant process (CRP) perspective, a restaurant exists for each type of n-gram. Therefore, a Chinese restaurant franchise with three branches is adopted in the model. In the unigram restaurant, each table serves a morpheme type (e.g. *book*); in the bigram restaurant, each table serves a morpheme bigram type (e.g. *book+ing*), and in the trigram restaurant each table serves a morpheme trigram type (e.g. *book+ing+s*). An illustration of the Chinese restaurant franchise is given in Figure 2.

IV. Inference

For the inference, we use Metropolis-Hastings algorithm [12]. In each iteration, we draw a word from the corpus and sample a segmentation from the following probability distribution:

$$p(w_j|D^{-w_j}, \alpha_u, \alpha_b, \alpha_t, \gamma, H_m, H_u, H_b, H_t) \quad (4)$$

where D^{-w_j} denotes the corpus excluding $w_j = m_1 + m_2 + \cdots + m_z$. A segmentation is sampled using the trigram language model given in Equation 1.

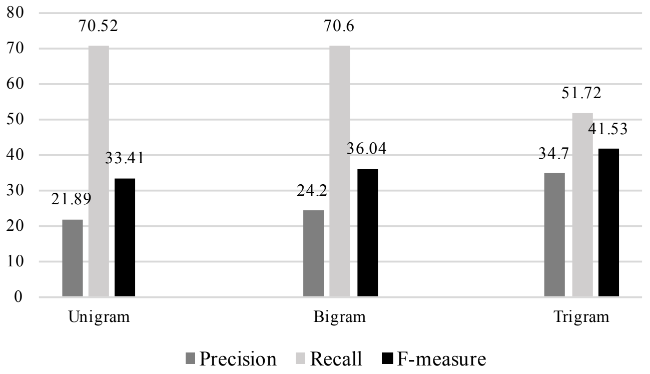

Fig. 3. The results on the Turkish dataset.

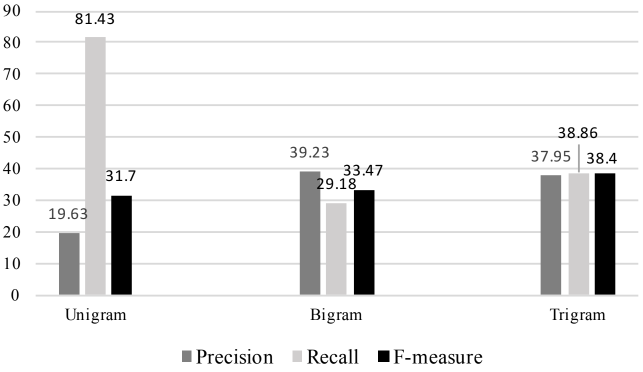

Fig. 4. The results on the Finnish dataset.

Let T denote the morpheme trigrams, B denote the bigram morphemes, and M denote the unigram morphemes in the model. The conditional probability of a trigram is computed as follows:

$$p(m_i|m_{i-1}, m_{i-2}, M^{-<m_i>}, B^{-<m_{i-1},m_i>},\\ T^{-<m_{i-2},m_{i-1},m_i>}, \alpha_b, \alpha_u, \alpha_t, \gamma, H_m, H_u, H_b, H_t) =\\ \begin{cases} \dfrac{n^{T^{-<m_{i-2},m_{i-1},m_i>}}_{<m_{i-2},m_{i-1},m_i>}}{n^{B^{-<m_{i-1},m_i>}}_{<m_{i-1},m_i>} + \alpha_t}, \\ \quad \text{if } < m_{i-2}, m_{i-1}, m_i > \in T^{-<m_{i-2},m_{i-1},m_i>} \\ \dfrac{\alpha_t * H_b(m_i|m_{i-1}, M^{-<m_i>}, B^{-<m_{i-1},m>}, \epsilon)}{n^{B^{-<m_{i-1},m_i>}}_{<m_{i-1},m_i>} + \alpha_t}, & \text{otherwise} \end{cases}$$

where ϵ denotes the set of parameters $\{\alpha_b, \alpha_u, \gamma, H_m, H_u\}$, $n^{T^{-<m_{i-2},m_{i-1},m_i>}}_{<m_{i-2},m_{i-1},m_i>}$ is the number of occurrences of trigram $< m_{i-2}, m_{i-1}, m_i >$ once this trigram instance is removed from the corpus and analogously $n^{B^{-<m_{i-1},m_i>}}_{<m_{i-1},m_i>}$ denotes the number of bigrams of type $< m_{i-1}, m_i >$ once the current instance of that type is removed.

$H_b(m_i|m_{i-1}, M^{-<m_i>}, B^{-<m_{i-1},m>}, \epsilon)$ is computed by using the second-level Dirichlet process as follows:

$$H_b(m_i|m_{i-1}, M^{-<m_i>}, B^{-<m_{i-1},m>}, \epsilon) =\\ \begin{cases} \dfrac{n^{B^{-<m_{i-1},m_i>}}_{<m_{i-1},m_i>}}{n^{M^{-<m_{i-1}>}}_{<m_{i-1}>} + \alpha_b}, \text{if } < m_{i-1}, m_i > \in B^{-<m_{i-1},m_i>} \\ \dfrac{\alpha_b * H_u(m_i|M^{-<m_i>}, H_m, \alpha_u, \gamma)}{n^{M^{-<m_{i-1}>}}_{<m_{i-1}>} + \alpha_b}, \text{otherwise} \end{cases}$$

where $n^{B^{-<m_{i-1},m_i>}}_{<m_{i-1},m_i>}$ is the number of occurrences of $< m_{i-1}, m_i >$, when this bigram instance is removed from the model.

Finally, $H_u(m_i|M^{-<m_i>}, H_m, \alpha_u, \gamma)$ is calculated by using the unigram Dirichlet process as follows:

$$p(m_i|M^{-<m_i>}, H_m, \alpha_u, \gamma) =\\ \begin{cases} \dfrac{n^{M^{-<m_i>}}_{<m_i>}}{N + \alpha_u}, & \text{if } < m_i > \in M^{-<m_i>} \\ \dfrac{\alpha_u * H_m(m_i)}{N + \alpha_u}, & \text{otherwise} \end{cases}$$

where N is the total number of morpheme tokens and $n^{M^{-<m_i>}}_{<m_i>}$ is the number of occurrences of $< m_i >$ once this morpheme token is excluded.

V. Evaluation and Results

We used publicly available Morpho Challenge [13] datasets for training and testing. We used the combined training and development sets for training without using the gold analyses. Gold analyses were used only for the evaluation. We did experiments on three different languages: English, Turkish, and Finnish. The combined gold sets involve 1686, 1760, and 1835 words for each language respectively.

We set the parameters $\alpha_u = \alpha_b = \alpha_t = \gamma = 0.01$ manually as a result of several experiments on each language.

We used Morpho Challenge evaluation method. For the precision, we draw two words from the results randomly and check whether they indeed share a common morpheme in the gold sets. For each correct morpheme pair, we assign one point. The total number of points is divided by the number of morpheme pairs compared. The same is also applied for recall, but this time by drawing a word pair from the gold sets and checking whether they indeed have a common morpheme in the results. The F-measure is the harmonic mean of precision and recall.

We did experiments by using morpheme unigrams, bigrams and trigrams in order to see the difference. The Turkish results are given in Figure 3, Finnish results in Figure 4, and English results in Figure 5. We can see that longer n-grams give better results. Turkish and Finnish are morphologically similar and both are agglutinating. English does not have a complex morphology as Turkish and Finnish, but still the results are quite similar when unigram, bigram and trigram models are compared. This supports the morpheme dependency even in languages with a simpler morphology.

We also compare our models with Morfessor Baseline [1] and Morfessor CatMAP [3]. The Turkish, Finnish and English, results are given in Table I, Table II, and Table III. The trigram model outperforms both Morfessor Baseline and Morfessor CATMAP with a F-measure of %41.53 on Turkish and %38.40 on Finnish. However, our models perform poorly on

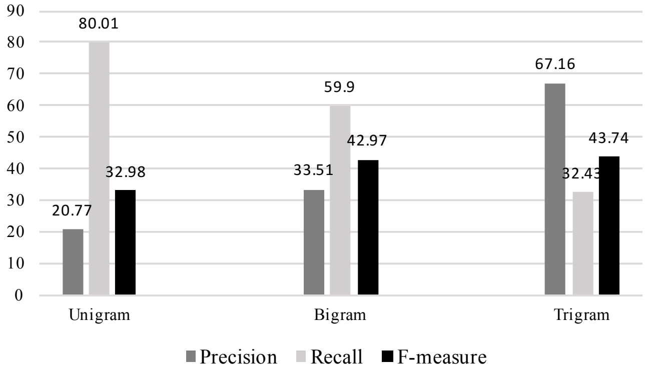

Fig. 5. The results on the English dataset.

TABLE I
COMPARISON OF OUR MODEL ON TURKISH

System	Precision (%)	Recall (%)	F-measure (%)
Unigram DP	21.89	70.52	33.41
Bigram HDP	24.20	70.60	36.04
Morfessor CatMAP [3]	85.72	25.09	38.81
Morfessor Baseline [1]	77.19	26.95	39.95
Trigram HDP	34.70	51.72	41.53

English when compared to Morfessor Baseline and Morfessor CatMAP.

An example of correct and incorrect segmentations in Turkish are given in Table IV. Due to sparsity in the datasets, the system cannot find some trigram patterns. However, it can find the bigrams and unigrams better than trigrams. A bigger training set would solve the sparsity issue in trigrams and would perform better.

VI. CONCLUSION

In this paper, we aim to show that morphemes are linked to each other and more dependency gets involved, better the system performs. We model morphology with morpheme trigrams by using a three-level HDP. Each level works as an interpolated smoothing for the next level by eliminating any sparsity in the n-grams.

Our results are far behind the state-of-art systems. However, our results shows that morphemes are dependent on each other as also implied by morphotactics. Hence, modeling morphemes jointly rather than independently improves the scores in morphological segmentation.

Dependency between morphemes has been scarcely worked in morphological segmentation in the literature. From this perspective, the paper fills an important gap in the literature by addressing the morpheme dependencies.

We tested our model on small sets due to complexity. Testing our models on bigger sets remain as a future work.

TABLE II
COMPARISON OF OUR MODEL ON FINNISH

System	Precision(%)	Recall(%)	F-measure(%)
Unigram DP	19.68	81.43	31.70
Bigram HDP	39.23	29.18	33.47
Morfessor Baseline [1]	71.94	23.93	35.91
Trigram HDP	37.95	38.86	38.40

TABLE III
COMPARISON OF OUR ON ENGLISH

System	Precision (%)	Recall (%)	F-measure (%)
Unigram DP	20.77	80.01	32.98
Bigram HDP	33.51	59.90	42.97
Trigram HDP	67.16	32.43	43.74
Morfessor Baseline [1]	79.15	53.89	64.12
Morfessor CatMAP [3]	82.63	51.19	63.21

TABLE IV
EXAMPLES OF CORRECT AND INCORRECT SEGMENTATIONS IN TURKISH

Correct	Incorrect
Onlem+i+dir	sevecen+li+kle (correct: sevecen+lik+le)
balkon+da+ki	fener+l+erin (correct: fener+ler+in)
kopya+lar+I	bozukluG+un (correct: boz+uk+luG+un)
prenses+le	bulut+l+ar+I+n (correct: bulut+lar+In)
zarf+I+nIn	baka+n+lara (correct: bakan+lar+a)
varis+ler+den	Onemse+me+li (correct: Onemse+meli)

REFERENCES

[1] M. Creutz and K. Lagus, “Unsupervised discovery of morphemes,” in *Proceedings of the ACL-02 workshop on Morphological and phonological learning - Volume 6*. Association for Computational Linguistics, 2002, pp. 21–30.

[2] M. Creutz and K. Lagus, “Unsupervised morpheme segmentation and morphology induction from text corpora using Morfessor 1.0,” *Technical Report A81*, 2005.

[3] M. Creutz and K. Lagus, “Inducing the morphological lexicon of a natural language from unannotated text,” in *Proceedings of the International and Interdisciplinary Conference on Adaptive Knowledge Representation and Reasoning*, 2005, pp. 106–113.

[4] S. Virpioja, O. Kohonen, and K. Lagus, “Unsupervised morpheme discovery with Allomorfessor,” in *Working Notes for the CLEF 2009 Workshop*, September 2009.

[5] S. Goldwater, M. Johnson, and T. L. Griffiths, “Interpolating between types and tokens by estimating power-law generators,” in *Advances in Neural Information Processing Systems 18*. MIT Press, 2006, pp. 459–466.

[6] B. Can and S. Manandhar, “Probabilistic hierarchical clustering of morphological paradigms,” in *Proceedings of the 13th Conference of the European Chapter of the Association for Computational Linguistics*, ser. EACL ’12. Association for Computational Linguistics, 2012, pp. 654–663.

[7] K. Narasimhan, R. Barzilay, and T. S. Jaakkola, “An unsupervised method for uncovering morphological chains,” *TACL*, vol. 3, pp. 157–167, 2015.

[8] T. Mikolov, K. Chen, G. Corrado, and J. Dean, “Efficient estimation of word representations in vector space,” *CoRR*, vol. abs/1301.3781, 2013.

[9] B. Snyder and R. Barzilay, “Unsupervised multilingual learning for morphological segmentation,” in *In The Annual Conference of the Association for Computational Linguistics*, 2008.

[10] B. Can, S. Kumyol, and C. Bozşahin, “Allomorphs and binary transitions reduce sparsity in turkish semi-supervised morphological processing,” in *Proceedings of the First Conference on Turkic Computational Linguistics (TurCLing)*, ser. TURCLing, 2016, pp. 49–54.

[11] M. Aksana, U. Demirhan, and Y. Aksan, “Corpus frequency and affix ordering in turkish,” in *Proceedings of the 17th International Conference on Intelligent Text Processing and Computational Linguistics (CICLing)*, April 2016.

[12] W. K. Hastings, “Monte Carlo sampling methods using Markov chains and their applications,” *Biometrika*, vol. 57, pp. 97–109, 1970.

[13] M. Kurimo, K. Lagus, S. Virpioja, and V. Turunen, “Morpho challenge 2010,” http://research.ics.tkk.fi/events/morphochallenge2010/, 2011, online; accessed 7-June-2016.