Synopsis: Study on Word Networks

Joy Deep Nath (03CS3021)

under the guidance of
Prof. Niloy Ganguly

Abstract content

A study of the word interaction networks of Bengali in the framework of complex networks is first done. The topological properties of these networks reveal interesting insights into the morpho-syntax of the language, whereas clustering helps in the induction of the natural word classes leading to a principled way of designing POS tagsets. Then, different network construction techniques and clustering algorithms based on the cohesiveness of the word clusters measured against two gold-standard tagsets by means of the novel metric of *tag-entropy*. The approach is then extended to any five other languages- English, German, Hindi, Hebrew and Finnish to find their word network properties. Since the culsters on manual inspection reveal word classes hinting at named entities being clustered, we perform named entity recognition on Hindi and Bangla and compare it against known results. And finally, we create a word network from tagged corpus of six languages to study the network structure and verify the morphosyntactic structure.

1. Introduction

Parts-of-speech (POS, also known as *word class* or *lexical category*) are the linguistic categories of words defined by their morphological and syntactic properties. The word categories that are distinctive in one language may feature identical behavior in another language. Linguists identify the lexical categories through a manual inspection of the morpho-syntactic patterns present in a language. Can there be a principled and computational approach to this problem of identification of the lexical categories? The answer turns out to be 'yes', thanks to the concept of "distributional hypothesis" (Harris(1968)).

In fact, this hypothesis is the underlying (implicit or explicit) assumption of all computational approaches to POS tagging which is a very important preprocessing task for several NLP applications. Ironically, compared to the work done in the area of POS tagging, the volume of research dedicated to POS tagset (i.e., the set of lexical categories) design is quite small, even though the tagset is largely responsible for the efficiency as well as the effectiveness of a POS tagger.

The two basic questions that need to be answered while designing a POS tagset are: (a) which lexical categories are distinguishable in a language? and (b) does making a distinction between two categories help us in further NLP applications such as chunking and parsing? In other words, a tagset is always dependent on the language under consideration as well as the end application to which the POS-tagger caters. In fact, often the natural word classes present in a language are those that are easy to distinguish as well as sufficient in facilitating deeper linguistic processing. A key to the identification of these natural word classes is to understand the syntactic structure of a language, which is captured through the complex interaction of the words. This is arguably an outcome of a self-organizing process governing the dynamics of language and grounded in the cognitive abilities of human beings (Steels(2000)). In this context, language can be viewed as a network of words and formation of lexical categories an emergent property of this network. Thus, understanding the structure and function of this network will help us in procuring deeper insight into the nature of word classes in a given language.

In this work, a study of the lexical classes of Bengali obtained through the analysis of the word interaction networks is presented. Although the scheme presented here is not essentially novel and has been motivated by several work on unsupervised induction of POS based on the distributional hypothesis (Finch and Chater(1992); Schütze(1993); Schütze(1995); Gauch and Futrelle(1994); Clark(2000); Rapp(2005); Biemann(2006b)), the main contributions reside in – (a) a comparative study of various approaches to POS tagset induction on Bengali, (b) rigorous linguistic analysis of the word classes and suggestions for a Bengali tagset design, (c) introduction of a novel metric, called tag entropy, to evaluate the goodness of the induced word classes, and most importantly, (d) analysis of the word interaction networks within the framework of complex network theory to understand the syntactic structure of Bengali, (e) extended the analytical scheme to five other languages (f) Creating a framework for analytical engine and (g) studying the word networks of tagged corpus for the 6 languages.

2. Word Networks

The definition and the construction of the word networks presented here are primarily based on the work by (Biemann(2006b)). Nevertheless, we also explore some variations while defining the network as well as their construction for Bengali data. Moreover, we study the topological properties of these networks, which provides us with insights into the syntactic structure of Bengali. We also conduct a comparative study of two different clustering algorithms.

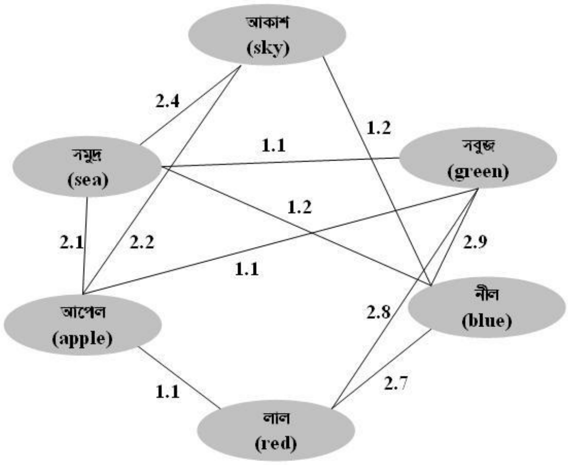

Figure 1: A hypothetical illustration of the word network. The English gloss for each of the Bengali words is provided within parentheses. Note that the edge weights are hypothetical and do not correspond to any of the similarity metrics.

2.1. Feature words, Context Vectors and Similarity Metric

We take a raw Bengali text corpus consisting of n tokens and compute the unigram frequency counts for each of the types observed in the corpus. We select the first m types that have the highest unigram frequencies as the *feature words*. The intuition is that since the function words have a very high frequency, the feature words selected on the basis of frequency will largely correspond to the function words of the language.

However, we observe that for corpora pertaining to specific domains (e.g., only news articles), several content words also creep into the list of top few words deemed here as feature words. Therefore, to ensure the absence of any content word in the set of feature words, we also construct networks where the this set is manually selected from a frequency-based sorted list of words. We shall refer to the former (i.e., frequency based feature word selection) networks by a prefixed superscript *fr* and the latter networks by another prefixed superscript *ms*.

Let $w_{-2}w_{-1}ww_1w_2$ be a window of 5 tokens around the target word w. A *context vector* for the target word w is defined as a vector of dimension $4m$ in which the entries $(4i+1), (4i+2), (4i+3)$ and $(4i+4)$ correspond to the number of occurrences of the $(i-1)$th feature word at the w_{-2}, w_{-1}, w_1 and w_2 positions respectively.

In (Biemann(2006b)), the distributional similarity between two words w and v is defined as $sim(w,v) = \frac{1}{1-cos(\vec{w},\vec{v})}$, where $\vec{w}$ and $\vec{v}$ represent the context vectors of the words w and v computed from a large raw text corpus; $cos(\vec{x},\vec{y})$ is the normalized dot product of the vectors $\vec{x}$ and $\vec{y}$, i.e., the cosine of the angle between them. An alternative definition of the similarity could be simply the cosine of the angle between $\vec{w}$ and $\vec{v}$, that is $sim(w,v) = cos(\vec{w},\vec{v})$. We shall denote the networks constructed using the metric proposed in (Biemann(2006b)) by a prefixed superscript *b* (for Biemann) and the latter ones by another prefixed superscript *c* (for cosine).

2.2. Definition and Construction of the Networks

The word network is a weighted undirected graph $G = \langle V, E \rangle$, where V consists of 5000 nodes corresponding to the most frequent 5000 types excluding the feature words. The number of nodes in V has been decided based on the fact that with a corpus of size around 10M words, enough context information is available only for the top few words. The weight of the edge between any two nodes representative of the words w and v is given by $sim(w,v)$ and this edge exists if $sim(w,v)$ exceeds a threshold τ. Thus, considering all the variations in definition of feature words and similarity metric, we can construct four different networks for a given corpus: ${}^{fr,b}G$, ${}^{fr,c}G$, ${}^{ms,b}G$ and ${}^{ms,c}G$.

Figure 1 presents a hypothetical illustration of the word network.

We have used the newspaper corpus[1] Ananda Bazaar Patrika for the creation of word networks. This corpus has around 17M words. We shall represent a network constructed from a corpus of size n using m feature words as $G_{n,m}$. Therefore, for a frequency-based selection of feature words and cosine similarity metric, the networks will be denoted as ${}^{fr,c}G_{n,m}$. Also, we shall drop the superscripts or subscripts whenever we refer to the networks corresponding to all the combinations for the part dropped.

We construct 20 word networks for all possible combinations of $n = \{$ 1M, 2M, 5M, 10M, 17M $\}$ and $m = \{25, 50, 100, 200\}$. In order to construct $G_{n,m}$ for $n < 17$, we have randomly selected a subset of documents from the original corpus. Note that in our experiments we consider the different inflected forms of a root morpheme as different types.

[1] Grateful to ISI Kolkata for providing this corpora for the purpose of the experiments.

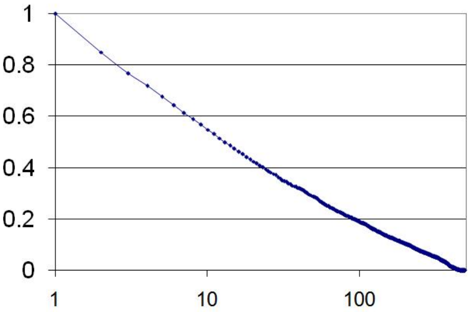

Figure 2: Cumulative Degree Distribution for the word network ${}^{fr,b}G_{17M,50}$. x-axis: $\log(k)$, y-axis: P_k

2.3. Properties of the Word Networks

In this section we present some of the important topological properties of the word networks. Interestingly, the four basic variations in network construction give rise to networks that have very similar topological properties. Therefore, we shall present all the results for ${}^{fr,b}G_{n,m}$, which might be generalized to the other cases as well. Note that the calculation of the degree distribution and the clustering coefficient is done on the unweighted version of the networks (all edges below the threshold τ are deleted).

2.3.1. Degree Distribution

The cumulative degree distribution (CDD) of a network, P_k, is the probability that a randomly chosen node has degree greater than or equal to k. CDD provides important information about the topology of the network. Figure 2 shows the CDD for the word network ${}^{fr,b}G_{17M,50}$. We observe that the CDD follows a logarithmic distribution (i.e., $P_k \propto log(k)$), which means that $-\frac{dP_k}{dk} = p_k$ (probability that a randomly chosen node has degree equal to k) or the non-cumulative degree distribution is proportional to k^{-1} (popularly known as power-law or Zipfian distribution, but it is not clear whether this is a consequence of Zipf's law). Similar results have been observed for the networks with varying m and n.

Power-law networks are believed to have a self-similar hierarchical structure. In this case, the hierarchy is a reflection of syntactic ambiguities. Highly ambiguous words that belong to several lexical categories have the highest degrees. The next level of hierarchy is manifested by words that belong to a few lexical categories, whereas the last level of hierarchy is represented by the words that are unambiguous in nature. The power-law indicates that there are few words that belong to a large number of lexical categories, while the most of the words belong to only one lexical category.

2.3.2. Clustering Coefficient

The clustering coefficient of a node is the probability that a randomly chosen pair of its neighbors are themselves neighbors. We observe that there is a positive correlation between the degree of a node and its clustering coefficient. In particular, high degree nodes (i.e., the most ambiguous ones) have a high clustering coefficient. This implies that the network is very dense (clique-ish) around the high degree nodes. As we shall see later, this has a significant effect on the cluster size distribution and the efficacy of this method as such. The mean clustering coefficient for ${}^{fr,b}G_{17M,50}$ is 0.53, which is much higher than that of random graphs. This again points to the fact that there is a strong community structure in the networks reflecting the presence of natural word classes.

2.4. Community Structure

In order to gain insight into the topology of the network we cluster them using the following two different approaches.

Chinese Whispers: The Chinese Whispers (CW) algorithm (Biemann(2006a)) is a non-parametric random-walk based clustering algorithm, where initially each node is in a separate cluster. In every iteration, the nodes propagate information about their current cluster to all the neighbors, and in turn, decide upon their own cluster labels based on a weighted majority voting of the cluster information received from the neighbors. The algorithm terminates when the labels do not change considerably over successive iterations.

Agglomerative Hierarchical Clustering: In this approach (Rapp(2005)), initially all the words are in separate clusters. At every iteration, two clusters closest to each other (where "closeness" between the centroids of the two clusters is measured by $sim(w, v)$) are merged to form a new cluster. The algorithm terminates after obtaining a predefined number of clusters.

We plot the cluster size distributions for ${}^{fr,b}G$ in Fig. 3 for various values of n and m following the CW algorithm. In fact, the distributions are identical for both the clustering approaches and all the other networks. The cluster size distributions (CSD) show a power-law behavior, which gets better as n increases. Thus, there are a few giant clusters, as is expected from the presence of the nodes with high degree and high clustering coefficient in the networks. Thus, the giant clusters consist of words that belong to multiple POS categories. In fact, these are the words that make POS tagging a non-trivial and challenging task. It would be interesting to devise techniques that can break the giant component into further clusters. We also observe that the words belonging to the giant clusters need not have high frequency in the corpus.

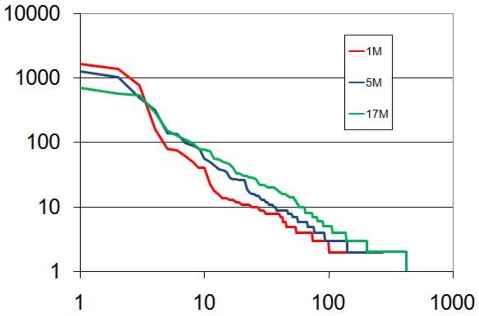

Figure 3: Rank (x-axis) versus cluster size (y-axis) in doubly logarithmic scale for ${}^{fr,b}G_{n,50}$ where n is 1M, 5M and 17M. The clusters are assigned a rank in descending order of their size (i.e. the number of words in the cluster), so that the largest cluster gets rank 1.

n	m	Baseline	$MTE(C)$	$WMTE(C)$	% gain for MTE	% gain for $WMTE$	m	Baseline	$MTE(C)$	$WMTE(C)$	% gain for MTE	% gain for $WMTE$
1M	25	4.09 (4.02)	1.75 (1.09)	3.51 (3.30)	57 (73)	14 (18)	100	4.10 (4.03)	1.61 (1.11)	3.57 (3.38)	61 (72)	13 (16)
	50	4.08 (4.01)	1.69 (1.10)	3.53 (3.32)	59 (72)	13 (17)	200	4.11 (4.05)	1.77 (1.12)	3.60 (3.44)	57 (72)	12 (15)
2M	25	4.13 (4.09)	1.60 (0.99)	3.48 (3.30)	61 (76)	16 (19)	100	4.12 (4.08)	1.56 (1.00)	3.51 (3.36)	62 (75)	15 (18)
	50	4.11 (4.08)	1.58 (1.01)	3.49 (3.31)	62 (75)	15 (19)	200	4.14 (4.10)	1.55 (0.96)	3.55 (3.39)	63 (76)	14 (17)
5M	25	4.08 (4.06)	1.52 (1.04)	3.23 (3.04)	63 (74)	21 (25)	100	4.04 (4.01)	1.46 (0.94)	3.22 (3.04)	64 (77)	20 (24)
	50	4.03 (4.01)	1.49 (0.95)	3.21 (3.04)	63 (76)	20 (24)	200	4.03 (4.01)	1.36 (0.81)	3.21 (3.05)	66 (80)	20 (24)
10M	25	4.06 (4.07)	1.41 (0.88)	3.16 (2.94)	65 (78)	22 (28)	100	4.08 (4.10)	1.35 (0.83)	3.17 (2.97)	67 (80)	22 (27)
	50	4.05 (4.07)	1.38 (0.88)	3.16 (2.95)	66 (78)	22 (28)	200	4.07 (4.09)	1.28 (0.77)	3.20 (3.04)	69 (81)	21 (25)
17M	25	4.04 (4.04)	1.53 (1.04)	3.03 (2.83)	62 (74)	25 (30)	100	3.96 (3.97)	1.38 (0.85)	2.97 (2.78)	65 (79)	25 (30)
	50	3.95 (3.96)	1.45 (0.99)	2.93 (2.74)	63 (75)	**26 (31)**	200	3.98 (3.99)	1.32 (0.76)	2.98 (2.81)	67 (81)	24 (29)

Table 1: Results for $CW_{n,m}$ model. The values in parentheses refer to the case where the words unknown to the morphological analyzer have been manually corrected. Best results in bold font.

In this section, we have analyzed the word networks from a complex network perspective, which has revealed several significant properties underlying the syntactic structure of Bengali. We shall revisit these issues in Sec. 8., but before that we shall analyze the word clusters from the perspectives of NLP and linguistics in general.

3. Experiments and Evaluation

Evaluation of the word clusters is challenging and there are two different ways in which this can be done. One way would be to compare the word clusters against a pre-designed set of lexical categories, in which case we are biased towards some gold standard tagset and consequently, contradicting the objective of automatic induction of the categories. Moreover, this method is incapable of evaluating the goodness of the clusters that are finer than the standard tagset. A better way is to resort to some task completion method for evaluation. Unfortunately, in absence of any standard task completion based evaluation strategy for the current work, we compare the clusters against two gold standard tagsets for Bengali described in (Dandapat et al.(2004)Dandapat, Sarkar, and Basu) and (Dasgupta and Ng(2007)).

3.1. Tag Entropy

Given a word w, a morphological analyzer returns all the possible segmentation of the word w along with the corresponding lexical categories[2]. For example, the Bengali word *kare* has three possible categories: NN (noun), gloss: palm - locative; VF (finite verb), gloss: do - present, simple, third person; and VN (non-finite verb), gloss: having done.

Let $cat_1, cat_2, \ldots cat_T$ be the universal set of lexical categories, where T is the total number of categories. We define a T-dimensional binary vector Tag_w for a word w as the *tag-vector*, where the value of $Tag_w(i)$ is 1 if and only if according to the morphological analyzer cat_i is a possible category for w. Thus, the tag-vector of *kare* will have 1 only in three positions (corresponding to the categories NN, VF and VN) and rest $T-3$ positions have 0s.

Given a cluster $c = \{w_1, w_2, \ldots w_s\}$, the cluster is perfectly cohesive if the tag-vectors of all the words in c are identical. On the other hand, the cluster is incohesive if the 1s and 0s are distributed randomly across them. Our objective is to define a metric over the tag vectors of the words in c, which will be able to quantify the cohesiveness of the cluster. Since binary entropy (Shannon and Weaver(1949)) measures the disorderedness of a system, we define the (in)cohesiveness of a cluster c of size s as

$$TE(c) = -\sum_{i=1}^{T} (p_i(c) \log_2 p_i(c) + q_i(c) \log_2 q_i(c)) \quad (1)$$

[2] For the tagset presented in (Dandapat et al.(2004)Dandapat, Sarkar, and Basu), we use the morphological analyzer for Bengali described in the same paper. However, for the purpose at hand, it suffices to have a lexicon with all the inflected forms of the root words and their categories. This is what we perform for the tagset presented in (Dasgupta and Ng(2007)).

where

$$p_i(c) = \frac{1}{s}[\text{\# words in } c \text{ for which } Tag_w(i) = 1]$$

and $q_i(c) = 1 - p_i(c)$.
In words, $TE(c)$ is the sum of the binary entropies of the cluster over each of the categories. We call $TE(c)$ the *tag entropy* of the cluster c. For a perfectly cohesive cluster, $p_i(c)$ is 1 or 0 for all i, and therefore, $TE(c) = 0$. For a perfectly incohesive cluster, $TE(c)$ is T. This happens when $p_i(c) = 0.5$ for all the categories. The lower the tag entropy, the higher the cohesiveness of the cluster.

3.2. Evaluation Metrics

The clustering algorithm splits the 5000 words into several clusters. Let $C = \{c_1, c_2, \ldots c_r\}$ be the set of word clusters for a particular experimental setup. Based on tag entropy, we define two metrics for evaluation of C: *mean tag entropy* $MTE(C)$ and *weighted mean tag entropy* $WMTE(C)$, as follows.

$$MTE(C) = \frac{1}{r}\sum_{i=1}^{r} TE(c_i) \tag{2}$$

$$WMTE(C) = \frac{1}{5000}\sum_{i=1}^{r} |c_i| TE(c_i) \tag{3}$$

where $|c_i|$ is the number of words in cluster c_i.
We define our baseline as the case when all the 5000 words are in the same cluster. Thus, the baseline MTE is equal to the baseline WMTE, which in turn is equal to $TE(V)$, where V is set of nodes in the network[3]. The motivation behind the definition of baseline is as follows. The quantity $TE(V) - WMTE(C)$ gives an estimate of information gain with respect to the standard tagset by splitting V into set of clusters C. Therefore, the higher the value of this quantity, the better the clustering.

3.3. Experiments

We use the 17M word *Anandabazaar Patrika* (a Bengali daily: http://www.anandabazar.com/) corpus for all our experiments. We have 4 different methods for network construction, 20 different combinations of m and n, 2 different clustering algorithms and 2 gold standard tagsets. This together gives rise to $4 \times 20 \times 2 \times 2 = 320$ possible experiments. It is quite a formidable task to report all these experiments here. Therefore, we divide our experiments into three sets, where we systematically investigate certain parameters.

3.3.1. Set I

In this set of experiments, we fix the network to ${}^{fr,b}G_{n,m}$, use CW clustering algorithm and compare our results for the (Dandapat et al.(2004)Dandapat, Sarkar, and Basu) tagset. Thus, we have 20 experiments corresponding to the various combinations of m and n, the results of which are summarized in Table 1. The aim of this set of experiments is to study the behavior of the clusters as we increase the corpus size and number of feature words. There are 450 to 500 clusters (including singletons) per graph found by the CW algorithm[4]. There were a large number of named entities among the target words that were unknown to the morphological analyzer. These words, around 1900 in number, have been manually assigned the appropriate POS categories and included for computation of WMTE.
The best results are obtained for $n = 17M$ and $m = 50$. As is expected, the goodness of the induced lexicon increases rather significantly with the corpus size. For a given corpus, using more feature words does not necessarily improve the results. In general, the ideal value of m seems to be a monotonically increasing function of n.

3.3.2. Set II

In this set of experiments, we investigate the effectiveness of the four different graph construction methods. For this set, we only use the hierarchical clustering method. The evaluations are made against the (Dandapat et al.(2004)Dandapat, Sarkar, and Basu) tagset and all the graphs are constructed for $n = 17M$ and $m = 50$, for which the best results are obtained in Set I.
The primary observation is that the hierarchical clustering gives better result than the CW algorithm. Nevertheless, unlike CW, the WMTE is lower (or the information gain is higher) for hierarchical clustering when the named entities are manually corrected. This implies that CW is able to cluster the named entities more efficiently than hierarchical clustering. Among the graph construction methods, the best results are obtained for ${}^{ms,c}G$, which shows that manual selection of feature words has a positive impact on the word clusters. This revalidates the fact that function words are better suited for POS tag induction.

[3] This is a slight abuse of notation because V is the set of nodes, whereas TE is defined on set of words. Nevertheless, the notation is unambiguous as every node in V correspond to one and only one word.

[4] Some of the example clusters can be found at *http://banglaposclusters.googlepages.com/home*

Metric	${}^{fr,b}G$	${}^{fr,c}G$	${}^{ms,b}G$	${}^{ms,c}G$
WMTE	36.2 (25.3)	37.7 (30.1)	36.7 (26.1)	**39.2 (38.1)**
MTE	86.7 (87.4)	64.0 (75.2)	**87.9 (88.9)**	70.5 (75.5)

Table 2: Percentage gain in MTE and WMTE for the 4 different graph construction and agglomerative hierarchical clustering. Best results are in bold fonts. The values in parentheses refer to the case where the words unknown to the morphological analyzer have been manually corrected.

3.3.3. Set III

As we have mentioned earlier, it is not appropriate to evaluate the goodness of the word clusters that emerge after clustering based on a predefined set of tags. One way to circumvent this problem is to evaluate across multiple tagsets. The previous two sets of experiments are based on the tagset defined in (Dandapat et al.(2004)Dandapat, Sarkar, and Basu). In the third set of experiments, we use the tagset described in (Dasgupta and Ng(2007)) and the dataset made available by the authors (*http://www.hlt.utdallas.edu/∼sajib/posDatasets.html*) consisting of 5000 Bengali words and their corresponding tags to evaluate our clusters. Since we do not have an access to the training corpus used in (Dasgupta and Ng(2007)), we have filtered our clusters obtained during the experiments in Set I and Set II, so that they contain only words present in the Dasgupta and Ng dataset. Consequently, the clustered networks now contain around 800 words.
The best results have been obtained for the combination of ${}^{fr,b}G_{17M,50}$ and CW algorithm, for which the entropy reduction is 89% and 57% for MTE and WMTE respectively. Note that these figures are 75% and 31% in the case of Dandapat et al. tagset. The best results for hierarchical clustering is obtained for ${}^{fr,c}G_{17M,50}$, where the respective reductions are 88% and 42%. Although it is tempting to reason that the vast improvement in the results for the Dasgupta and Ng dataset is because of the small number of tags, in reality this might not be the case as the baseline tag entropies for both the datasets are close (around 4). In the next section, we shall discuss the possible reasons behind this improvement.

4. Linguistic Analysis and Tagset Design

Bengali is an Indo-Aryan language spoken in Bangladesh and the eastern parts of India. The syntax of the language is morphologically rich and the word order is relatively free. The case relations between the verb and its arguments are usually marked by inflectional suffixes on the nouns. There are a handful of overloaded suffixes that mark various cases depending on the context. Verbs inflect for tense, aspect, mood and person. There are three non-finite verb forms that act as participles and gerund. Bengali has a small repertoire of verb roots and a large number of compound verbs are formed by noun-verb and adjective-verb combinations. Use of "do-support" verbs are also extremely common. Bengali makes use of classifiers (a word/morpheme used to classify nouns according to meaning, number, definiteness etc.), but does not distinguish between gender. Although number distinctions are sometimes reflected through nominal classifiers or suffixes, it is not marked on the verbs.
There has been very few work towards POS tagging in Bengali and consequently there are no standard and well-accepted tagset for the language. For instance, the two tagsets that we have used as gold standards differ substantially in their design principles. The tagset presented in (Dandapat et al.(2004)Dandapat, Sarkar, and Basu) has 40 tags covering the nouns (2 classes), verbs (6 classes), adjectives and quantifiers (6 classes), pronouns (11 classes) and other function words. This tagset is heavily influenced by the English Penn Treebank tagset and words are tagged primarily based on their syntactic function, rather than morphological form. Thus, except for the verbs, the different morphological variations of a root word are not placed into different lexical categories. On the other hand, the tagset described in (Dasgupta and Ng(2007)) consists of only 11 tags that partially covers the lexical categories of Bengali. Nouns are divided into 7 classes based on proper vs. common, singular vs. plural and different case-marker (genitive, locative, accusative and nominative) distinctions. There is one class each for adjectives and adverbs. Verbs are divided into two classes based on their morphological form (finite or non-finite). Hence, this tagset has been designed based on the forms of the words rather than their functions.
Let us investigate the nature of the clusters that emerged during our experiments. As discussed earlier, in all the experiments we observe the presence of a few (typically 2 to 4) giant clusters that mainly consist of ambiguous words and thus are "bad" clusters. In fact, it has been observed that by filtering the top few large clusters one can considerably reduce the tag entropy of the clustering. Manual inspection reveals that the medium to small size clusters are "good" and mostly composed of words belonging to similar morpho-syntactic category. There are, however, a few clusters formed on the basis of semantic similarity between the constituent words. See Table 4. for some example clusters[5].
The trends in which clusters are formed and merged during the hierarchical clustering provides us useful information about the distinguishabilty between the various lexical classes. We enumerate some of the natural classes that emerged out of our experiments and the categorical distinctions that seem needless for Bengali.
Nouns: Possessive nouns and pronouns (e.g. *gharera* 'of house', *tomAra* 'your') form a separate cluster and are similar to adjectives in their distribution than other nouns. Although nouns with locative (e.g. *ghare* 'in house') and accusative (e.g.

[5] In this article, we use Romanized script to represent Bengali words following the ITRANS (*http://www.aczoom.com/itrans/*) convention.

Size	Example Words	Remarks
596	*aruNa, buddhabAbu, saurabha, rAkesha, siddhArtha*	Proper nouns (names of person)
352	*golamAlera* 'of problem', *dAbira* 'of demand', *phalera* 'of result', *Agunera* 'of fire', *dUShaNera* 'of pollution'	Nouns with possessive marker
133	*badalAno* 'to change', *AmAnya* 'disregard', *AkramaNa* 'attack', *sAhAyya* 'help',*guli* 'bullet',	Nouns/verbal nouns that form compound verbs with 'do' or 'be'
44	*sAtaTi* 'seven', *tinaTe* 'three', *anekguli* 'many', *3Ti* 'three', *11Ti* 'eleven'	Quantifiers (mainly cardinal)
13	*adhibeshane* 'during the session', *bhAShaNe* 'in the speech', *baktRRitAYa* 'in the speech', *dalei* 'in the party', *pratibedane* 'in a report'	A semantic cluster related to parliamentary affairs

Table 3: Examples of clusters from the $^{fr,b}G_{17M,50}$ using CW algorithm.

Language	Corpus Size (in sentences)	Clustering Coefficient
Bengali	0.5 M	0.533
English	6.0 M	0.449
Finnish	11.0 M	0.469
German	40.0 M	0.486
Hebrew	1.7 M	0.498
Hindi	2.5 M	0.522

Table 4: Clustering Coefficients of word networks of six languages. All the networks were created using Chinese Whispers Clustering Algorithm with the 10,000 target words and 200 features

pradhAnamantrIke 'to the prime minister') case-markers form separate clusters initially, they merge with other nouns at a later stage of clustering. We further observe that there is no distinction between the distributions of plural and singular nouns.

Proper Nouns: Different clusters emerge for the different types of proper nouns, such as names of person, location, organization, month and days. Moreover, first and last names of persons show up as separate clusters.

Verbs: In all the models we observe that finite (e.g. *kareChena* 'have done'), modal (e.g. *pAre* 'can do'), non-finite (e.g. *uThe* 'having stood up') and infinitive (e.g. *karate* 'to do') verbs emerge as four basic categories. Non-finites and infinitives merge at a later stage. Verbal nouns (e.g. *khAoyA* 'to eat') form a separate cluster initially and later merge with nouns.

Adjectives and Numbers: The distinctions between quantifiers, intensifiers and numbers are observable, though in the later stages of clustering the former two categories merge with other adjectives.

Other Categories: We also observe the question words (e.g. *kI* 'what', *kemana* 'how'), relative pronouns (e.g. *ye* 'whoever', *yakhana* 'whenever'), punctuation marks, conjuncts (e.g. *o* 'and', *bA* 'or') forming separate clusters. However, since these are closed-classes with a very few representative words, it is difficult to make any strong claims about their naturalness.

Therefore, one should take into account the aforementioned factors while designing a tagset for Bengali. Despite the fact that the tagset of (Dasgupta and Ng(2007)) makes a larger number of distinctions between the noun forms, this partial tagset, as reflected in our experiments in Set III, has a better correlation with the natural word classes obtained. On the other hand, the Dandapat *et al.* tagset scores poorly on this dimension, primarily because of the finer distinctions made for the verbs and pronouns based on their function. Nevertheless, advanced stages of NLP like chunking and other applications might require such finer distinctions that are not apparent from the natural word classes. This, infact suggests that this clustering property could possibly helps us annotate large amounts of data and hints at a NER framework mentioned in Sec. 6..

5. Other word networks

The generic analytical framework is then used on five more languages viz. English, Finnish, German, Hebrew and Hindi to obtain the degree distribution, clustering coefficient of these languages. The clustering coefficients the six languages are shown in Table 4. Also, the cumulative degree distributions of the word netwroks for the six languages, it is evident from the values that word networks share similar structure. And hence, the framework is indeed applicable to any other languages.

6. Application to NER

Since we found on maunal inspection of the clusters that the natural word classes are being captured in the clusters, we design a framework to get a semi-supervised method to obtain Named Entity Recognition (NER). We start from the clusters obtained in the end of Chinese Whispers (we could have chosen Agglomerative too). We then manually identify the clusters which comtain the top 20 names of persons (NPE) and top 20 names of locations (NPL) label each word in those clusters to NPE and NPL respectively. We find that medium-sized clusters in which the members of the top 20 NPE/NPL occur are mostly pure; i.e. they tend to contain other words which are also NPE or NPL respectively. As a result, using (a) list of 20 names of places,(b) list of 20 names of locations and (c) the word clusters obtained from our experiments as inputs we obtain a larger augmented list of names of locations and places.

Using list of names of locations and places, we tag the whole corpus as NPE if the word is in the augmented list of names of persons; NPL if the word is in the augmented list of locations and NN otherwise. For each of the words we also obtain features like - Part-of-Speech (POS) tag using a standard tagger for the word, previous two words and the next two words. Using these features fore each word, we learn a NER using a CF-tree to obtain a NER tagger. This framework gives us a semi-supervised NER engine for the language.

7. Network of tagged words

Inspired by the strong associativity of POS tags to clusters derived from the word network in one hand and existence of clusters that are associated with the multiple POS tags, we define a new network- POS-word network. We start with a tagged corpus. Each and every word will possibly have multiple tags. If T=$<t_1,\ t_2,\ t_3,\ t_4,\ ...\ ,\ t_k>$ is a set of all possible tags, then each word $w_i$in the corpus would have a feature vector V_i=$<n_i1,\ n_i2,\ n_i3,\ n_i4,\ ...\ ,\ n_ik>$, where each of $n_ij's$ correspond to the number of times the word w_i gets the tag t_j. Now, taking the top 10,000 words as nodes and cosine distance between the words as the edge weights (we ofcourse threshold the edges), we obtain a weighted word network to observe that, indeed the ambiguous words have high degrees and clustering coefficients.

8. Conclusion

In this work, we presented a principled and systematic approach to understand the syntactic structure of Bengali and induce the natural word classes of this language. We summarize below our salient observations.

- The degree distribution of the network follows a power-law behavior reflecting a hierarchy of the words with respect to their syntactic ambiguities.

- The clustering coefficient of the network is significantly higher than that of the random graphs pointing to the presence of strong community structures that are representative of the natural word classes.

- Clustering splits the network into word classes representing different lexical categories and the cluster size distribution follows a power-law. There are a very few giant clusters consisting of many ambiguous words and a large number of medium to small size clusters consisting of mostly unambiguous words.

- The results obtained for all the different graph construction and clustering algorithms are very close to each other implying the underlying robustness of the distributional hypothesis. However, the size of the corpus has a strong effect on the quality of the emerging clusters.

- We note that morphology plays a significant role in defining the syntactic clusters of Bengali. However, it may be harmful to start with the assumption proposed in (Dasgupta and Ng(2007)) that each morphological category defines a syntactic class. In particular, we do observe possessive nouns and finite, non-finite and infinitive verbs forming separate clusters, but we also observe that presence of plural markers (e.g. *der*, *rA*) or accusative or locative inflections for nouns need not essentially mark a separate syntactic category.

In conclusion, the pen and paper based linguistic analysis technique for identification of lexical categories might well be automated in a principled manner by exploiting the concept of distributional hypothesis. Cross-linguistic study of the topology of the word networks can reveal several universal properties as well as typological variations in the linguistic systems. Apart from providing insights into the natural word classes leading to the design of appropriate tagsets, the study of these networks can significantly increase our understanding of the evolution of syntax as we study the word network.

9. References

Biemann, C., 2006a. Chinese whispers - an efficient graph clustering algorithm and its application to natural language processing problems. In: Proceedings of HLT-NAACL'06 workshop on TextGraphs. pp. 73–80.
URL http://www.aclweb.org/anthology/W/W06/W06-3812

Biemann, C., 2006b. Unsupervised part-of-speech tagging employing efficient graph clustering. In: Proceedings of COLING/ACL'06 Student Research Workshop. pp. 7–12.

Clark, A., 2000. Inducing syntactic categories by context distribution clustering. In: Cardie, C., Daelemans, W., Nédellec, C., Sang, E. T. K. (Eds.), Proceedings of CoNLL/LLL'00. pp. 91–94.
URL citeseer.ist.psu.edu/clark00inducing.html

Dandapat, S., Sarkar, S., Basu, A., 2004. A hybrid model for parts-of-speech tagging and its application to Bengali. International Journal of Information Technology 1 (4), 169–173.

Dasgupta, S., Ng, V., 2007. Unsupervised part-of-speech acquisition for resource-scarce languages. In: EMNLP-CoNLL'07. pp. 218–227.
URL http://www.aclweb.org/anthology/D/D07/D07-1023

Finch, S., Chater, N., 1992. Bootstrapping syntactic categories using statistical methods. In: Background and Experiments in Machine Learning of Natural Language: Proceedings of the 1st SHOE Workshop. Katholieke Universiteit, Brabant, Holland, pp. 229–235.

Gauch, S., Futrelle, R., 1994. Experiments in Automatic Word Class and Word Sense Identification for Information Retrieval. In: Proceedings of the 3rd Annual Symposium on Document Analysis and Information Retrieval. pp. 425–434.

Harris, Z. S., 1968. Mathematical Structures of Language. Wiley.

Rapp, R., 2005. A practical solution to the problem of automatic part-of-speech induction from text. In: Proceedings of ACL'05 (companion volume). pp. 77 – 80.

Schütze, H., 1993. Part-of-speech induction from scratch. In: Proceedings of ACL'93. pp. 251–258.

Schütze, H., 1995. Distributional part-of-speech tagging. In: Proceedings of EACL'95. Morgan Kaufmann Publishers Inc., San Francisco, CA, USA, pp. 141–148.

Shannon, C. E., Weaver, W., 1949. The Mathematical Theory of Information. University of Illinois Press.

Steels, L., 2000. Language as a complex adaptive system. In: Proceedings of PPSN VI. pp. 17–26.