Detecting the Presence of Named Entities in Bengali: Corpus and Experiments

Farzana Rashid
University of North Carolina at Asheville
frashid@unca.edu

Fahmida Hamid
New College of Florida
fahmida.hamid@gmail.com

Abstract

Named Entity Recognition (NER) belongs to the field of Information Extraction (IE) and Natural Language Processing (NLP). NER aims to find and categorize named entities present in the textual data into recognizable classes. Named entities play vital roles in other related fields like question-answering, relationship extraction, and machine translation. Researchers have done a significant amount of work (e.g., dataset construction and analysis) in this direction for several languages like English, Spanish, Chinese, Russian, Arabic, to name a few. We do not find a comparable amount of work for several South-Asian languages like Bengali/Bangla. Hence, as part of the initial phase, we have constructed a qualitative dataset in Bengali.

In this paper, we identify the presence of Named Entities (NEs) in the Bengali text (sentences), classify them in standardized categories, and test whether an automatic detection of NE is possible. We present a new corpus and experimental results. Our dataset, annotated by multiple humans, shows promising results (F-measures ranging from 0.72 to 0.84) in different setups (support vector machine (SVM) setups with simple language features and Long-Short Term Memory (LSTM) setup with various word embedding).

Introduction

Named Entity Recognition (NER) is a highly sought-after tool to facilitate several tasks like Machine Translation, Question Answering, Information Retrieval, and Natural Language Processing. But NER has not been very easy for several reasons. We list a few of the reasons below:

1. Named entities belong to the open class; i.e., people create new terms as named entities.
2. Some words can represent names as well as other types, given different contexts. Example:
   (a) We saw armies marching towards the **Square**$_{\textbf{loc}}$.
   (b) Please draw a **square** on the paper. (Here, the **square** refers to a square-shaped object; hence, it is not identified as an NE.)
3. Detecting the types of NEs can sometimes be very difficult due to the lack of context or ambiguous usage. Example:
   (a) I have a plan to meet with **Wendy's**$_{\textbf{person}}$ boyfriend.
   (b) I have a plan to buy lunch from **Wendy's**$_{\textbf{org/loc}}$.
4. We use numbers to indicate the cardinality of things or the time of a day. Example:
   (a) There are **eight**$_{\textbf{cardinal}}$ apples.
   (b) I will be there at **eight**$_{\textbf{time}}$.

Despite several challenges, a considerable amount of work has been done in NER for English and some other languages. Researchers are working hard to build and improve resources and tools for other low-resource languages. Unfortunately, Bengali/Bangla, the mother tongue of more than 220 million people, is one of the low-resource languages. Hence, we attempt to build a qualitative corpus in Bengali that can be used for named entity recognition. In this work, we present the corpus and provide experimental reports to show that the corpus can be reliably used to detect the presence of the named entities.

NER in Bengali

Every language has its beauty and challenges. Bengali is no exception. Besides some of the issues mentioned in the previous section for English, Bengali has some unique challenges (Ekbal and Bandyopadhyay 2010). We present a few here:

1. Unlike English, Bengali lacks capitalization information (as shown in the sentences in figure 1) which helps in NE detection.
2. Bengali is a low-resource language lacking in enough annotated corpora, gazetteers or good POS-taggers etc.
3. Bengali is neither entirely agglutinative nor entirely fusional. Morphemes do not always have clear boundaries in this language. A single affix may conflate multiple morphemes. The statement holds for named entities as it holds for verbs and other classes. We frequently see some accusative, genitives, locatives, and adjectives attached right after the named entities. We see a few examples in figure 1.

Bengali sentences with their English translations	NEs in their pure forms with their types, and the alternative forms
ইত্যাদি$_{event}$ করিমের$_{person}$ প্রিয় শো। English: Ettaydi is Karim’s favorite show.	event: ইত্যাদি (Ettaydi) person: করিম (Karim) করিম + এর = করিমের <genitives>
করিমদের$_{person}$ বাসায়$_{loc}$ ভীষণ গোলমাল চলছে। English: A chaos is happening at Karim’s house.	person: করিম (Karim) loc: বাসা (house) করিম + দের = করিমদের <genitive> বাসা + য় = বাসায় <locative>
যমুনা সেতুটি$_{LOC}$ বাংলাদেশের$_{gpe}$ একটি গর্ব। English: The Jamuna bridge is a pride of Bangladesh.	loc: যমুনা সেতু (Jamuna bridge) gpe: বাংলাদেশ (Bangladesh) বাংলাদেশ + এর = বাংলাদেশের <genitive> সেতু + টি = সেতুটি <adjective>
ভোরের$_{time}$ আকাশ দেখতে কি যে ভালো লাগে! English: How wonderful is it to watch the morning sky!	time: ভোর (morning) ভোর + এর = ভোরের <genitive>
শেফালীকে$_{person}$ তার ঊর্ধ্বতন কর্মকর্তা জরুরি তবল করলেন। English: Shefali was called immediately by her boss.	Person: শেফালী(Shefali) শেফালী + কে = শেফালিকে < accusative>

Figure 1: The sentences show how accusatives, genitives, locatives, and adjectives come attached right after the NE’s.

4. Names in Bengali are quite diverse; thus, it adds to the ambiguity factor.
5. Technological advancement in Bengali NLP is still at its young age.

	κ
Has NE or not	0.97
Numbers of NE	0.91
Types of NE	0.91

Table 1: Inter-annotator agreements (shown as as κ coefficients) for whether there is a presence of NE, the number of NEs and the types of NEs. κ values between 0.6 and 0.8 indicate *substantial* agreement, κ values over 0.8 indicate *nearly perfect* agreement (Artstein and Poesio 2008).

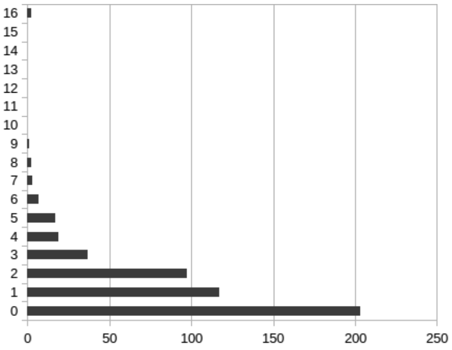


Figure 2: Frequencies of NEs in sentences. The vertical axis show the number of NE’s per sentence, and the horizontal axis shows the number of sentences that have a particular number of NE’s.

Data and Annotation

In order to build the corpus, we needed a good number of Bengali sentences that would potentially contain a wide variety of name entities (NE) of several types. We also needed Bengali sentences containing no NE so that we could form a balanced dataset to run a classifier that could detect the presence and absence of NEs.

We collected 503 target sentences from several articles from a few popular Bengali daily newspapers along with their immediate previous sentence in the article and the immediate next sentence for context information. These articles covered a variety of topics including sports, entertainment, rural news etc. As the sentences are collected from newspaper articles, they are all written in formal Bengali and are well edited and contains factual information. The sentences were collected from several issues published in the second half of the year 2020. The sentences were carefully hand-picked by two native speakers of Bengali language who has researcher level knowledge in NLP. They collected the sentences after thorough reading of the articles.

These sentences were then made ready for annotations by the two native speakers who had originally collected them.

Annotation Process

At first both the annotators separately marked whether a target sentence contained any NE or not. Once both agreed on the categorization of sentences in the two sets, they took the set of sentences containing the NEs for the second round of annotations. They identified each of the NE and marked them with the type and the position numbers of the tokens that formed the NEs keeping in mind the previous and the next sentences as context information.

After approximately 10% of the data were annotated, both the annotators carried out adjudication to discuss any disparity in their labeling. All the sentences were annotated by both the annotators with high agreements. Table 1 shows agreements on whether a sentence contains NE, how many NEs are there and on the types of the NE’s. The Cohen’s κ coefficients of these agreements are $>.90$ which is substantially high. κ values between 0.6 and 0.8 indicate *substantial* agreement, κ values over 0.8 indicate *nearly perfect* agreement (Artstein and Poesio 2008). After another round of adjudication, the best agreed upon annotations were finalized.

Annotation Statistics

Figure 2 shows the frequency’s of NE’s in sentences. 203 sentences out of the 503 did not have any presence NE in them. About 117 sentences had exactly one NE and two sentences had 16 NE mentions each. It is evident from fig 2 that few sentences had large numbers of NE’s and most sentences had one, two or three NE’s. There is no sentence containing 10-15 NE’s.

Figure 3 shows the distribution of the different types of NE’s. The most common type of NE present in our dataset is type PERSON (315 out of the total 703). NE’s of types ORG and GPE fill the next two positions appearing 90 and

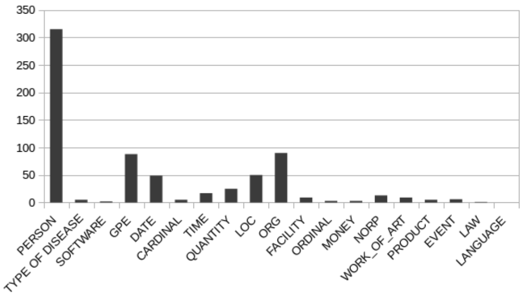


Figure 3: The distribution of different types of NEs. The horizontal axis show the types of NEs present in our dataset, and the verticle axis shows the number of instances of each type of NE.

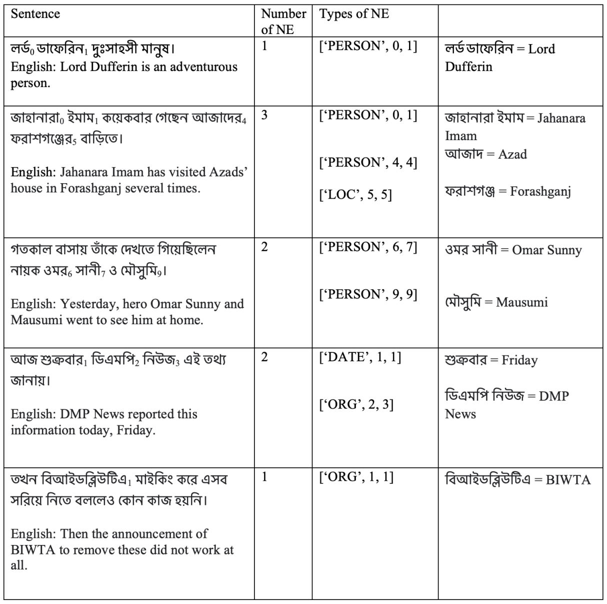
		P	R	F
	Majority baseline	.36	.60	.45
SVM	first word	.57	.59	.56
	+ length range	.62	.59	.60
	+BOW	.68	.68	.68
	+TFIDF	.72	.72	.72
NN	LSTM (GloVe)	.76	.75	.76
	LSTM (Keras)	.84	.84	.84

Table 2: Results (Precisions(P), Recalls(R) and F-measures (F)) of SVM setups with different feature sets and of NN with LSTM setups with Bengali GloVe embeddings (Sarker 2020) and Keras Embeddings.

88 times respectively. Types LOC and DATE also appear quite a few times. Given the domain of our data sources (i.e. news articles), this was to be expected. We also see a wide variety of NE's found in a corpus of about 500 sentences. The label distribution is studied to get an idea of the presence of different categories.

Annotation Examples

Figure 4 shows a few sample sentences that we annotated. The target sentences are shown along with the NE's in color and their token positions as subscripts. They are presented with the number of NE's contained and the types in the next two columns. Bengali sentences, just like those in any other language, contain mixed types of NE's.



Sentence	Number of NE	Types of NE	
লর্ড$_0$ ডাফেরিন$_1$ দুঃসাহসী মানুষ। English: Lord Dufferin is an adventurous person.	1	[‘PERSON’, 0, 1]	লর্ড ডাফেরিন = Lord Dufferin
জাহানারা$_0$ ইমাম$_1$ কয়েকবার গেছেন আজাদের$_4$ ফরাশগঞ্জের$_5$ বাড়িতে। English: Jahanara Imam has visited Azads' house in Forashganj several times.	3	[‘PERSON’, 0, 1] [‘PERSON’, 4, 4] [‘LOC’, 5, 5]	জাহানারা ইমাম = Jahanara Imam আজাদ = Azad ফরাশগঞ্জ = Forashganj
গতকাল বাসায় তাঁকে দেখতে গিয়েছিলেন নায়ক ওমর$_6$ সানী$_7$ ও মৌসুমি$_9$। English: Yesterday, hero Omar Sunny and Mausumi went to see him at home.	2	[‘PERSON’, 6, 7] [‘PERSON’, 9, 9]	ওমর সানী = Omar Sunny মৌসুমি = Mausumi
আজ শুক্রবার$_1$ ডিএমপি$_2$ নিউজ$_3$ এই তথ্য জানায়। English: DMP News reported this information today, Friday.	2	[‘DATE’, 1, 1] [‘ORG’, 2, 3]	শুক্রবার = Friday ডিএমপি নিউজ = DMP News
তখন বিআইডব্লিউটিএ$_1$ মাইকিং করে এসব সরিয়ে নিতে বললেও কোন কাজ হয়নি। English: Then the announcement of BIWTA to remove these did not work at all.	1	[‘ORG’, 1, 1]	বিআইডব্লিউটিএ = BIWTA

Figure 4: A few examples of our annotations. The numbers of NEs and the types and token positions (the subscripted numbers in the first column and the pair of numbers beside the types in the second column indicate the positions of the tokens) of each NE.

Experiments and Results

We use SVM classifiers with different combinations of features and neural network based LSTM classifier to classify the sentences into categories indicating the contained NE's or not.

SVM setup

We use SVM classifiers with RBF kernel to predict if a sentence contained NE or not. We divided the entire corpus into stratified train and test splits (80-20), and used the implementation in scikit-learn (Pedregosa et al. 2011) to train the classifiers. We tuned SVM hyperparameters (C and γ) using 10-fold cross-validation with the train split. Our classifier predicted the two labels 0 (no NE present) and 1 (at least 1 NE present).

Feature Set. We extract features from the target sentence only. The feature set does not include information about any other external information like the context sentences which were also collected alongside the target sentences.

The features set is rather simple and includes the first word in a sentence (first word), the range of the length of a sentence (length range), the bag-of-words representations (BOW), and the tf-idf representations (TFIDF) of the sentence. Classifiers trained with these features yield results which are very good as shown in table 2 despite their simplicity.

Neural Network Setup

We use two LSTM based setups for learning with Neural Network. The difference lies in the types of embeddings used.

For the first setup, as the first layer, we use the embedding layer offered by Keras. The layer works with a vocabulary of size 3270 (unique words in our dataset), a vector space of 300 dimensions in which words will be embedded, and input documents that have 50 words each. The output from the embedding layer will be 50 vectors of 300 dimensions each, one for each word. We pass this to an LSTM network (with 100 hidden nodes, 0.2 dropout) and finally we pass the output of the LSTM to the Dense output layer. For the dense layer, we use sigmoid activation function. For the model, we use binary cross entropy as the loss function and the adam optimizer. With 40% validation split, we train the model for 50 epochs.

We run a similar model as the above setup except for the embedding layer which uses weights coming from a 300 dimensional pre-trained word vector for Bengali words (Sarker 2020). We train the model for 15 epochs.

Results

The results of the SVM setups as well as the NN setups are summarized in table 2.

With the SVM setup, any combination of features outperforms the majority baseline (F-measure: 0.45). Using as features the first word increases the F-measure to 0.56 which is farther increased when the length range is included as a feature. Bag-of-words representations for the sentence brings the F-measure to 0.68. The tfidf representations yields 0.72 F-measure and thus including it in the feature set proves to be beneficial.

The NN setup that uses weights coming from a 300 dimensional pre-trained word vector for Bengali words yields an F-measure of 0.76; the setup that uses the embedding layer offered by Keras yields a much higher F-measure of 0.84.

Related Works

Even though many NER systems have been built for English, just a handful has been built for Bengali. NER systems for Bengali and Hindi using Support Vector Machine (SVM) was builtin 2010 (Ekbal and Bandyopadhyay 2010). A Hidden Markov Model based system (Ekbal and Bandyopadhyay 2007) and a system using Hidden Markov Model merged with rule-base approaches (Drovo et al. 2019) are there too for Bengali and other languages. There are language independent NER systems based on statistical Conditional Random Fields (CRFs) for a few South and South-east Asian languages (Ekbal et al. 2008) and also for just Bengali and Hindi (Ekbal and Bandyopadhyay 2009).

A corpus based on Bengali newspaper taken from web-archives is developed alongside an NER system based on pattern based shallow parsing with or without using linguistic knowledge (Ekbal and Bandyopadhyay 2008).

Unlike others, our aim was to build language independent systems based on SVM and LSTM for detecting the presence of NE's in Bengali text which can be used to automatically sort Bengali sentences that contain NE's to be later used for further annotations for corpus building. We also created a corpus with detailed annotations regarding NE's.

Conclusion

This paper focuses on the detection of the presence of named entities in Bengali textual data. We collected 500+ standard Bengali sentences from several leading newspapers of Bangladesh. We labeled the entities with a set of fine-grained universal NE tags (PERSON, ORGANIZATION, GPE, LOCATION, TIME, DATE, QUANTITY, CARDINAL, MONEY, etc.). Our current dataset has 19 distinct labels. Two NLP researchers carefully double annotated the sentences. The agreements of the annotations are promisingly high, an indication of a reliable and qualitative dataset. Our experiments show modest performances (both the SVM and LSTM based experiments). Future works in a similar domain may use them as baselines.

We plan to expand our work in several directions. First, we want to increase the corpus size without compromising the quality. As expected, we need extreme care and trained annotators to do so. We require a significant amount of time and other relevant resources to fulfill this plan. Second, we want to use our corpus for efficiently detecting the positions of the named entities in the sentences. Third, we plan to use the corpus to identify the types of relationships between multiple named entities present in a sentence. Others, researching Bengali textual data (or multilingual data), may also be able to use it in various ways. We release our annotations at http://www.cs.unca.edu/~frashid/datasets.

References

Artstein, R., and Poesio, M. 2008. Inter-coder agreement for computational linguistics. *Comput. Linguist.* 34(4):555–596.

Drovo, M. D.; Chowdhury, M.; Uday, S. I.; and Das, A. K. 2019. Named entity recognition in Bengali text using merged hidden markov model and rule base approach. In *2019 7th International Conference on Smart Computing Communications (ICSCC)*, 1–5.

Ekbal, A., and Bandyopadhyay, S. 2007. A hidden markov model based named entity recognition system: Bengali and Hindi as case studies. In Ghosh, A.; De, R. K.; and Pal, S. K., eds., *Pattern Recognition and Machine Intelligence*, 545–552. Berlin, Heidelberg: Springer Berlin Heidelberg.

Ekbal, A., and Bandyopadhyay, S. 2008. A web-based Bengali news corpus for named entity recognition. *Language Resources and Evaluation* 42:173–182.

Ekbal, A., and Bandyopadhyay, S. 2009. A conditional random field approach for named entity recognition in bengali and hindi. *Linguistic Issues in Language Technology* 2(1):1–44.

Ekbal, A., and Bandyopadhyay, S. 2010. Named entity recognition using support vector machine: A language independent approach. *International Journal of Electrical and Computer Engineering* 4(3):589 – 604.

Ekbal, A.; Haque, R.; Das, A.; Poka, V.; and Bandyopadhyay, S. 2008. Language independent named entity recognition in indian languages. In *Proceedings of the IJCNLP-08 Workshop on Named Entity Recognition for South and South East Asian Languages*.

Pedregosa, F.; Varoquaux, G.; Gramfort, A.; Michel, V.; Thirion, B.; Grisel, O.; Blondel, M.; Prettenhofer, P.; Weiss, R.; Dubourg, V.; Vanderplas, J.; Passos, A.; Cournapeau, D.; Brucher, M.; Perrot, M.; and Duchesnay, E. 2011. Scikit-learn: Machine learning in Python. *Journal of Machine Learning Research* 12:2825–2830.

Sarker, S. 2020. Bengali GloVe pretrained word vector. https://github.com/sagorbrur/GloVe-Bengali.