A Crowdsource-Based Approach for Preparing Bangla POS Tagged Corpus

Shamim Ehsan, Sadia Tasnim Swarna and Sabir Ismail

Abstract Automated Parts of Speech Tagging plays a vital role in the natural language processing. For computational Bangla Language Processing, we do not have large-scale Parts of Speech tagged corpus. There are two basic approaches to implement a corpus, by written rules or automated. To implement a rule-based corpus, we need experts in Bangla linguistics and it is also time-consuming. And for the automated corpus, we need a trained corpus, which is currently not available. Crowdsourcing can be served a vital role to fulfill these two requirements. So, in this paper, we proposed a crowd source-based approach to building Bangla Parts of Speech tagged corpus. We have used a standard tag set for Bangla. Raw documents are collected from various newspapers, books, and online site. We first give some example of Parts of Speech and then provide data to people for crowdsourcing. Finally, we analyze the result of the data, and its accuracy is 95%.

1 Introduction

Crowdsourcing is the act of taking a job traditionally performed by a designated employee and outsourcing it to an undefined, generally large group of people in the form of an open call. So basically its like someone has a task to be done, but do not want to hire a specific person to do it, rather they put their task on the world wide web with a small amount of reward, let random people solve it and finally getting the task done. This is how crowd sourcing works.

Crowdsourcing can be used in Parts of Speech Tagging. All words in sentences are classified into some groups which have some common grammatical properties; they are called parts of speech. Generally, the words having same parts of speech

S. Ehsan (✉) · S. T. Swarna (✉) · S. Ismail
Shahjalal University of Science and Technology, Sylhet 3114, Bangladesh
e-mail: ehsanhrid@gmail.com

S. T. Swarna
e-mail: sadiatasnimswarna@gmail.com

S. Ismail
e-mail: sabir-cse@sust.edu

A. Abraham et al. (eds.), *Emerging Technologies in Data Mining and Information Security*, Advances in Intelligent Systems and Computing 814,
https://doi.org/10.1007/978-981-13-1501-5_40

play similar roles inside the linguistic structure of sentences. Parts of Speech tagging means a system tags the parts of speech of a sentence automatically. It may be assumed that it can be easy for a computer by just looking into the dictionary for the appropriate parts of speech. But a word can fall into different parts of speech in the context of different sentences. Consider the sentences:

1. শরীরে বল নাই?
2. কথা বল।
3. বল খেলা শরীরের জন্য ভাল।

It is clearly seen that the word " বল " has three different meaning as well as different parts of speeches in the four sentences (adjective, verb, and noun). Unfortunately, there is no fixed rules to say " বল " belongs to which parts of speech in the sentences. To solve this ambiguity, we either need specific rules or we can train machine, to solve it using learning. Here comes the factor of data. Parts of speech tagging system needs a training dataset with all the sentences tagged with their corresponding parts of speech. But this kind of dataset is not available for Bangla.

So what to do? Then comes crowd sourcing with the solution. Crowdsourcing does not need very skilled workers or very sophisticated devices, rather a crowdsourcing system needs only a large number of normal people with no special talents/abilities.

Everyone with a minimum grammatical knowledge can tag sentence, and it does not take many skills. So tagging sentences using the general crowd can be a pretty good option to create a huge dataset of tagged sentences. It is pretty effective also in the fields of natural language processing like speech recognition, translation. Translation feature of Google Inc[1] is heavily depended on crowdsourcing from users all around the world, and they claim better accuracy than typical statistical machine translation. So we can say that crowdsourcing can solve the problem of automated parts of speech tagging.

2 Related Works

Crowdsourcing systems enlist a large number of humans to solve a wide variety of problems. Crowdsourcing was proposed to make efficient use of manpower and resources. In recent decade, numerous work has concentrated on various aspects of crowdsourcing, say for different performance analysis and computational techniques. Figure 1 shows a taxonomy of crowdsourcing.

Natural language processing is a work that may be difficult for automated process but comparatively easy for humans. In recent times, as a quick alternative of expert annotations, researchers found Mechanical Turk of Amazon [1–6]. Akkaya et al. [7] demonstrated that crowdsourcing would be good for subjectivity word sense comment. Callison-Burch and Dredze [8] demonstrated that they make information for discourse and dialect applications with a minimal effort. Gao and Vogel [3] showed that crowdsourcing laborers perform well on word arrangement assignments.

[1] https://translate.google.com/.

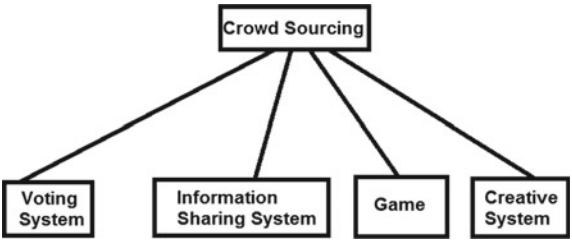

Fig. 1 A taxonomy of crowdsourcing

Jha et al. [4] demonstrated that an exact prepositional expression connection corpus can be developed by crowdsourcing specialists. Parent and Eskenazi [5] proposed a system to decompose a task for the meaning of dictionary words in MTurk. Skory and Eskenazi [6] discussed ways to calculate the quality of the results of MTurk workers tasks.

3 Raw Data Collection

First step is to select a group of sentence as our primary dataset. We cannot just choose some random sentence and ask users to tag them. Team Pipilikas[2] research team has shared their text corpus with us with 1000 documents in the corpus, all of which are collected from various newspapers. From there 330 documents are chosen. All the documents have passed through a validator then again manually checked to be selected for the dataset. The validator checks for too long and incomplete sentences. As user satisfaction should be the main priority in a crowdsourcing system, so if the user gets bored with very large sentences, that will prevent them from tagging more documents. After formatting, 330 documents have taken from PIPILIKA corpus, 35 documents from the novel " আমার বন্ধু রাশেদ " [9] By Dr. Muhammad Zafar Iqbal, and 35 from facebook articles from famous writers. So in total 400 documents have been chosen, each containing three sentences as our primary dataset. A brief summary is shown in Table 1.

Though there are five tags in Bangla grammar and eight parts of speech in English grammar, 12 tags are globally used in the parts of speech tagging research [10, 11]. They are:

1. বিশেষ্য(NN)
2. সর্বনাম(PR)
3. নির্দেশক(DM)
4. ক্রিয়া(V)
5. বিশেষণ(JJ)
6. ক্রিয়া-বিশেষণ(RB)
7. অনুসর্গ(PSP)
8. সংযোজক(CC)
9. অব্যয়(RP)
10. পরিমাণবাচক(QT)
11. যতি-চিহ্ন(PNC)
12. অন্যান্য(RD)

[2]Pipilika is the first Bangla search engine developed by the students of Shahjalal University of Science and Technology.

Table 1 Summary of the dataset

Total documents	400
Total sentences	1200
Total words	7657
Unique words	3082
Most frequent word	এ

A reference document is also used to help out the crowd workers. Niladri Sekhar Dashs [10] POS tagset for Bangla document is used as a reference document for crowd workers.

4 User Interface Implementation

4.1 Crowd Sourcing System

A system is called a crowdsourcing system if it depends on the opinions and answers of the general crowd. People who are participating in the crowdsourcing are called crowd users. Some crowdsourcing system sets some questions and takes opinion of the users, and most voted opinion is considered to be the correct answer. These systems are called Voting System. In some cases, there is no fixed answer to the question; users have to use their creativity to crowdsource. For example, Google Draw [12] ask users to draw an object within 20 s. Suppose a user is asked to draw a car, he has to draw it with his own creativity, these systems are called creative systems.

Our proposed crowdsourcing system is both a voting system and creative system. There will be sentences without any proper tags, users will tag the sentences, and with the response of the users, the model will tag the words.

4.2 Building the Interface

An interface is created to collect data from crowd workers. It's a web application which is built using PHP Laravel 5.3 framework. Laravel is chosen because it can handle concurrent requests smartly than most other frameworks. No pre-built web template is chosen, we tried to keep the site as simple as possible. Most of the templates have to load lots of JavaScript and CSS files and loading all of the files in every page makes the site a bit slower.

The user interface has register page, login page, and home page. Some people don't get motivated to enter when it requires too much information to log into a site,

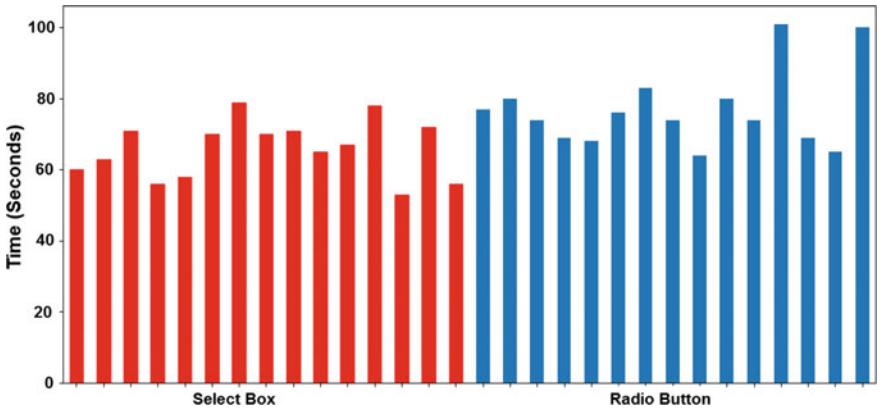

Fig. 2 Average time to tag a document in HTML select box and radio button

so a guest login feature where people get logged in with just a temporary username. After logging in, each user is assigned with a document and asked to tag the words of the documents.

We've planned to make use three different types of input fields of HTML (select box, radio button, and drag and drop). We successfully implemented first two and tested it to know which option will be suitable for the users. We select fifteen volunteers and assigned each of them to tag 50 sentences with solutions as fast as they can to check which method is faster and has less chance of selecting a wrong tag while in a hurry. Results are shown in Fig. 2.

It is seen that using select box, people take an average of 65 s and using radio button they take 76 s to tag a document. Using select box, people tag 98.45% words correctly while radio button has an accuracy of 91.45%. So we can have a conclusion that both chances of making silly mistakes and speed in radio button is higher than a select box. So we have excluded radio button from our site and take input from users with only HTML Select Box. When a user submits a document a confirmation window appears and the user is prompted with a view of the document with the corresponding parts of speech it has been tagged, if user noticed that he/she have tagged a word wrongly, he/she can move back to the previous page and correct it. In this case, the user doesn't have to select all the tags again, because if that happens, nobody wants to do that again. Instead, the user gets a view where all of his previous selected sentences is tagged and he just has to change the wrong tag and confirm. It is shown in Fig. 3.

4.3 Adding a LeaderBoard

As it is mentioned earlier, people only crowd-sourced when they get a handy reward or got tricked as they are competing or playing a game. So, a leaderboard is added where everyone can see who is at the top of data tagging. The user can see total

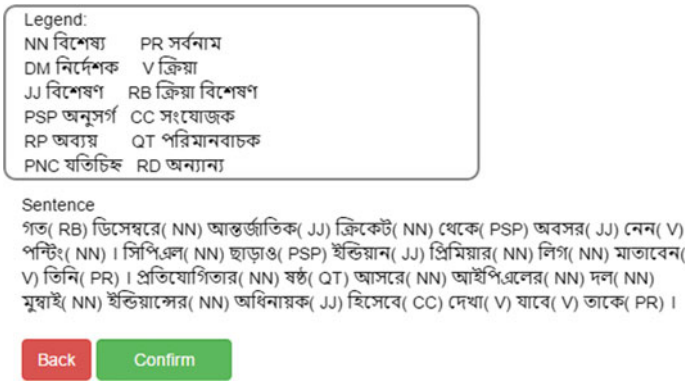

Fig. 3 Snapshot of Confirmation window

tagged documents region-wise. After adding the leaderboard feature, the average time to tag a document reduces more than 10% than the time takes before. Before adding leaderboard feature, the average time to tag a document by a user is 100 s, and after adding leaderboard, it reduced to 92 s.

5 Methodology

After the interface is created, data is collected from the crowd workers. Several people tag a sentence differently, and next step is to find out which tag should belong to which word in sentences, i.e., label the sentences. Like voting systems, we go for the majority. If the majority of people says that "ভাল" from "সে ভাল আছে" has the tag "বিশেষণ", then we label "ভাল" as "বিশেষণ" in the sentence "সে ভাল আছে". In case of a tie between two tags, we search for previous behavior of the word. Consider the sentence: "গত অর্থবছর থেকে আমদানিতে স্থবিরতা দেখা দেয়". Seven crowd workers have tagged the document and each of their tags have been shown in a 2-D matrix form. We can see that majority of people voted "অথর্বছর", "আমদানিতে" as "বিশেষ্য", "দেখা", "দেয়" as "ক্রিয়া" but confusion arises for the words গত and স্থিবরতা. In this case, we check previously in how many times "গত" has been tagged as "নির্দেশক" and as ক্রিয়া-বিশেষণ. Previous behavior breaks the tie. Matrix of the sentence with the most likely tag of: "গত অর্থবছর থেকে আমদানিতে স্থবিরতা দেখা দেয়" is shown in Fig. 4.

6 Finding Optimal Number of Users

As we have mentioned earlier, each document is tagged around 5–8 times. A question arises that how many times a document should be tagged? Five times should be enough? Is tagging a document eight times necessary? Or is it overkill? So the raw

	গত	অর্থবছর	থেকে	আমদানিতে	স্থবিরতা	দেখা	দেয়
বিশেষ্য	1	5	0	4	1	0	0
সর্বনাম	0	0	0	0	0	0	0
নির্দেশক	2	0	0	0	0	0	0
ক্রিয়া	0	0	0	2	0	7	6
বিশেষণ	1	2	0	1	3	0	0
ক্রিয়া-বিশেষণ	2	0	0	0	3	0	0
অনুসর্গ	0	0	5	0	0	0	1
সংযোজক	0	0	0	0	0	0	0
অব্যয়	1	0	2	0	0	0	0
পরিমাণবাচক	0	0	0	0	0	0	0
যতি-চিহ্ন	0	0	0	0	0	0	0
অন্যান্য	0	0	0	0	0	0	0
Most Likely Tag	নির্দেশক	বিশেষ্য	অনুসর্গ	বিশেষ্য	বিশেষণ	ক্রিয়া	ক্রিয়া

Fig. 4 Most Likely Tags of a sample sentence

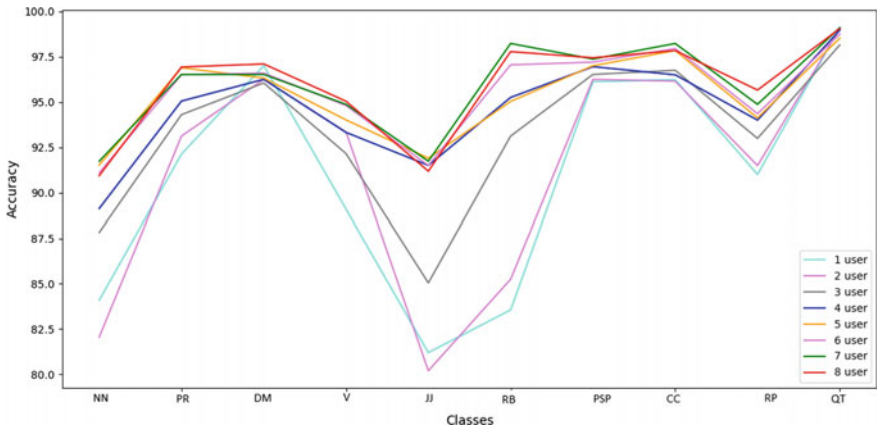

Fig. 5 Accuracy of different dataset to find optimal number of user to tag a document

dataset is analyzed again, and some tests are run. Eight data corpuses are created, each dataset is created based on the profile matrix of random n crowd workers response. If n exceeds the total number of responses to that document, all the responses to that document are taken. For the eight data corpuses, accuracy for each of the classes is calculated. The procedure of getting accuracy of dataset is discussed in Results section. The results are shown in Fig. 5. By observing the figure, we can see that when $n > 5$, accuracy for different parts of speeches is very slightly increasing, and in some cases, decreasing. So we can say that tagging a document by five different users should be enough.

7 Results

Total 170 crowd workers have been participated to build the tagged corpus, age between 20 and 25, all of them are a university student. On average, each worker tags 13 documents. Each document is tagged at least five times, and at most eight times. A total of 2195 times the documents has been tagged by the crowd workers. Each document contains three sentences, so in total 6585 times, the sentences are tagged. Table 2 shows a brief summary in a tabular form.

We run a test to see how many words have been tagged by all the users the same every time. Among the words which have been tagged same by all the users, a pie chart is given to show which parts of speech have been tagged more correctly in Fig. 6. The chart shows that tagging "বিশেষ্য" (Noun) and "ক্রিয়া" (Verb) are comparatively easier than other documents.

For testing, 10% of the random document is chosen and sent to some Bangla linguistic experts for checking. Then the accuracy is calculated for each of the classes. Confusion matrix of the classes True Positive, True Negative, False Positive, False Negative scores for the confusion matrix and the accuracy precision and recall are shown in Tables 3, 4, and 5.

Table 2 Summary of tagged documents

Total documents tagged	2195 times
Total sentences tagged	6585 times
Crowd worker	170
Average document tagged	5.48
Maximum number of times a document is tagged	8
Minimum number of times a document is tagged	5

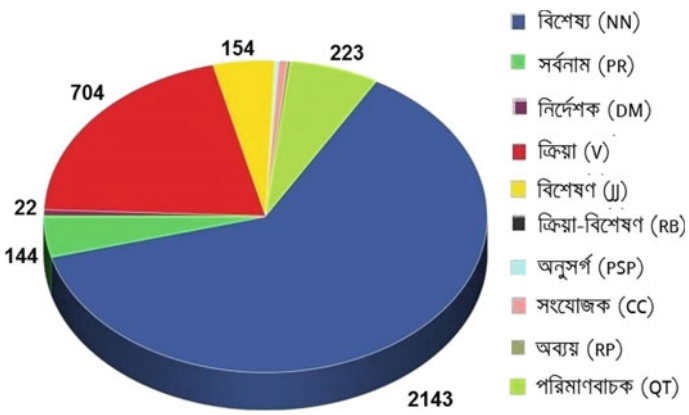

Fig. 6 Correctness of the parts of speeches

Table 3 Confusion matrix

	NN	PR	DM	V	JJ	RB	PSP	CC	RP	QT
NN	655	10	15	12	51	10	2	1	0	13
PR	5	221	0	0	15	0	0	0	0	0
DM	5	20	50	0	0	2	7	3	0	0
V	19	3	0	261	9	0	15	0	0	0
JJ	11	0	0	11	223	10	0	15	10	0
RB	0	0	0	0	15	26	0	0	0	0
PSP	0	0	0	0	0	0	24	0	10	0
CC	0	0	3	0	0	3	0	30	0	0
RP	13	0	0	0	15	0	5	15	40	0
QT	0	0	0	0	0	0	0	0	3	144

Table 4 True Positive, True Negative, False Positive, and False Negative table for confusion matrix

	TP	TN	FP	FN
NN	655	1019	114	53
PR	221	1453	20	33
DM	50	1624	37	18
V	261	1413	4	23
JJ	223	1451	57	105
RB	26	1648	15	25
PSP	24	1680	10	29
CC	30	1644	6	34
RP	40	1634	48	23
QT	144	1530	3	13

Table 5 Accuracy, Precision, and Recall of Test Dataset

	Accuracy	Precision	Recall
NN	90.93	85.18	92.51
PR	96.93	91.70	87.01
DM	96.82	57.47	73.53
V	96.04	85.02	91.90
JJ	91.18	79.64	67.99
RB	97.66	63.41	50.98
PSP	97.73	70.58	45.28
CC	97.67	83.33	46.88
RP	95.93	45.45	63.50
QT	99.05	97.96	91.72

A surprising thing is observed that the accuracy of the noun (বিশেষ্য) is a bit lower than the other classes. By common sense, we can assume that tagging noun is easier. But if we can look at the confusion matrix, we can see that many words have been falsely classified as a noun by the crowd workers, but they belong to some other class, Crowd workers tend to tag a document as noun when they are confused with the class of that word. That is the reason behind the lower accuracy of noun. The poor performance of Precision and Recall in some classes can also be explained; they seldom occur in the sentences, so a misclassification can damage more to Recall and Precision than Accuracy because in calculating accuracy, True Negative is also accounted. As all the True Negative values are comparatively larger, so a small value in False Positive and False Negative affects the accuracy less. The average accuracy for all the classes is 95.994%. So after calculating tags of every sentence according to our methodology stated above, our final tagged corpus is created. It is a table in MySQL with these rows:

- ID
- line
- tags
- category

Category field is kept for further research.

References

1. Quinn, A.J., Bederson, B.B.: Human computation: a survey and taxonomy of a growing field. In: Proceedings of the SIGCHI conference on human factors in computing systems. ACM (2011)
2. Gordon, J., Van Durme, B., Schubert, L.K.: Evaluation of commonsense knowledge with Mechanical Turk. In: Proceedings of the NAACL HLT 2010 Workshop on Creating Speech and Language Data with Amazon's Mechanical Turk. Association for Computational Linguistics (2010)
3. Gao, Q., Vogel, S.: Consensus versus expertise: a case study of word alignment with mechanical turk. In: Proceedings of the NAACL HLT 2010 Workshop on Creating Speech and Language Data with Amazon's Mechanical Turk. Association for Computational Linguistics (2010)
4. Jha, M., et al.: Corpus creation for new genres: a crowdsourced approach to PP attachment. In: Proceedings of the NAACL HLT 2010 Workshop on Creating Speech and Language Data with Amazon's Mechanical Turk. Association for Computational Linguistics (2010)
5. Parent, G., Eskenazi, M.: Clustering dictionary definitions using amazon mechanical turk.. In: Proceedings of the NAACL HLT 2010 Workshop on Creating Speech and Language Data with Amazon's Mechanical Turk. Association for Computational Linguistics (2010)
6. Skory, A., Eskenazi, M.: Predicting cloze task quality for vocabulary training. In: Proceedings of the NAACL HLT 2010 Fifth Workshop on Innovative Use of NLP for Building Educational Applications. Association for Computational Linguistics (2010)
7. Akkaya, C., et al.: Amazon mechanical turk for subjectivity word sense disambiguation. In: Proceedings of the NAACL HLT 2010 Workshop on Creating Speech and Language Data with Amazon's Mechanical Turk. Association for Computational Linguistics (2010)
8. Callison-Burch, C., Dredze, M.: Creating speech and language data with Amazon's Mechanical Turk. In: Proceedings of the NAACL HLT 2010 Workshop on Creating Speech and Language Data with Amazon's Mechanical Turk. Association for Computational Linguistics (2010)

9. Dr. Muhammad Zafar Iqbal, Rashed: My Friend, ISBN-984-437046-9
10. Niladri Sekhar Dash, POS tagset for Bangla Document, Microsoft Research India, Aug 2010
11. Categorizing and Tagging Words. http://www.nltk.org/book/ch05.html Cited 30 Aug 2017
12. Quick, Draw! https://quickdraw.withgoogle.com/, cited 30 Aug 2017