Combining Confidence Score and Mal-rule Filters for Automatic Creation of Bangla Error Corpus: Grammar Checker Perspective

Bibekananda Kundu[1,2], Sutanu Chakraborti[2], and Sanjay Kumar Choudhury[1]

[1] Language Technology, Centre for Development of Advance Computing,
Kolkata-700091, India
[2] Department of Computer Science and Engineering, Indian Institution of Technology,
Chennai-600036, India
{bibekananda.kundu,sanjay.choudhury}@cdac.in,
sutanuc@cse.iitm.ac.in

Abstract. This paper describes a novel approach for automatic creation of Bangla error corpus for training and evaluation of grammar checker systems. The procedure begins with automatic creation of large number of erroneous sentences from a set of grammatically correct sentences. A statistical Confidence Score Filter has been implemented to select proper samples from the generated erroneous sentences such that sentences with less probable word sequences get lower confidence score and vice versa. Rule based Mal-rule filter with HMM based semi-supervised POS tagger has been used to collect the sentences having improper tag sequences. Combination of these two filters ensures the robustness of the proposed approach such that no valid construction is getting selected within the synthetically generated error corpus. Though the present work focuses on the most frequent grammatical errors in Bangla written text, detail taxonomy of grammatical errors in Bangla is also presented here, with an aim to increase the coverage of the error corpus in future. The proposed approach is language independent and could be easily applied for creating similar corpora in other languages.

Keywords: Automatic Error Corpora Creation, Confidence Score, Mal-rule, Grammar Checking.

1 Introduction

Socrates's famous dictum was “Correct language is the prerequisite for correct living”. In the context of our everyday use of editing environments, the need of automatic grammatical error detection and correction cannot be overemphasized. The system plays a pivotal role in Computer-Assisted Language Learning (CALL) for second language learners. Its function can be also encapsulated as a post processor component of Machine Translation (MT) and Optical Character Recognition (OCR) system. One of the major limitations of using rule-based parser is the knowledge

A. Gelbukh (Ed.): CICLing 2012, Part II, LNCS 7182, pp. 462–477, 2012.

acquisition bottleneck and the inability to reliably capture the syntactic structure of free word order language like Bangla using Context Free Grammar rules. To the best of our knowledge, till now there is no robust rule-based parser is available for Bangla language. This observation has motivated elegant probabilistic and statistical interpretation of free word order languages. It also inspired a great deal of attention towards learning syntax from completely unannotated text. But most of the existing empirical error detection models have been hampered by unavailability of sufficiently large annotated learner's error corpora. There is a dearth of annotated error learner corpora of Bangla text depending on learner's age variation and social and educational influences. One of the major problem of building error corpus from learners' data is that the process is very time consuming and required linguistic knowledge to examine each sentence of learners' text to determine nature and density of errors. To overcome this problem, a corpus of ungrammatical Bangla sentences has been created automatically considering performance errors and language learning errors that occur frequently. This paper is more closely aligned to the task of automatic error corpora creation and does not focus on the methodology of an actual grammar checking system that can be built using the corpus. Before starting our discussion on automated error corpus creation methodology, we provide a background on the origin and linguistic aspects of Bangla language and illustrate types of text error of Bangla Second Language Learners at the time of writing text.

2 Background

Bangla is the fifth popular language in the world and the second in India. It is the national language of Bangladesh. This language belongs to the Indo-Aryan family and originated from Prakit which is a sister language of Sanskrit. Sister languages of Bangla are Oriya, Magahi and Maithili in the west and Assamese in the north east of India. Bengali and Assamese are the eastern most languages of the Indo-European family of languages. When compared to languages like English, Bangla is largely free from words orders with some specific limitations. Like other Asian languages it follows a Subject-Object-Verb (S-O-V) pattern but orientation of these three atoms is flexible, i.e. S-V-O is allowable but not popularly used. Inspite of these free movements there is an invisible bonding between words having a mutual attraction towards each other which is governed by the property "Valency".

2.1 Errors in Text

It has been seen that many people are fluent in speaking Bangla language but their writing skill is appalling because of their lack of grammatical knowledge of the language and oversight in the time of writing. Even professional writers occasionally succumb to such errors. Bangla Second Language Learners often commit grammatical mistakes while writing text because of their lack of language knowledge (Language Learning Error) and due to oversight, carelessness or tiredness (performance error). Performance errors can occur mainly due to four operations: insertion, deletion, transposition and substitution. When an error involves more than one operation, it is known as Composite Error. There are two primary concerns at the time of automatic error corpus creation, first one being linguistically realistic and the second one is to mimic the error scenarios that happen normally. To analyse the kind of naturally produced error scenario we have collected 1500 sentences from 10 standard native students' exam papers of Bangla and also have collected second language learners' data from students whose first language is either Hindi or Oriya or Telegu. Performance errors and language learning errors occurred in their text are then carefully analysed. Exam papers are collected with the assumption that students make more mistakes in the time of examination as they are usually in a hurry to complete their answers within the limited time period. In the course of studying Second Language Learners text, it has been found that the proportion of errors occurred by substitution operation is much more than any other operations. Figure 1 shows the proportion of performance errors caused by each of the four operations.

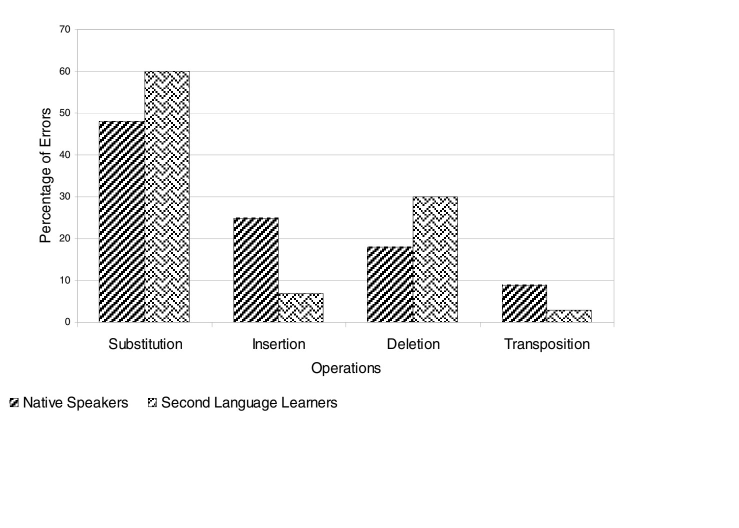


Fig. 1. Proportion of Errors in Native Speakers and Second Language Learners Corpus

The Native Speakers and the Second Language Learners make same kinds of mistakes such as misuse of punctuation and cohort/homophones [12]. But study shows that Second Language Learners make much more mistakes than native speakers. Most frequent error types produced by native speakers may not be produced by second language learners. For example, errors generated while writing complex sentences are infrequent for language learners, as most of the time language learners avoid writing complex sentences. They write complex sentences only when they have enough confidence in their ability to construct them correctly. Second Language Learners can be of two types viz. L1 and L2. Kind of errors produced by L1 Language Learners are influenced by their native language. When native languages are similar but not identical, L1 produces errors due to negative transfers. They fail to find exact equivalence between these two languages. On the other hand, L2 Language Learners produce errors because of their incomplete knowledge of syntactic and/or morphological irregularities. They face trouble due to the novelty of the new language [12]. After analyzing the collected Bangla second language learners' data we came to know that the above statements (quoted in [12]) are also true for Bangla language. Therefore, learners who learn Bangla language having the background of Oriya, Assamese or Hindi as native language produces different kinds of errors than learners having native languages like Malayalam, Tamil, Telegu or English. We have classified the types of errors according to the operations involved in performance error and also depending on language learning errors. We shall now elaborate below different kind of errors depicted by second language learners.

1. Transposition Operation:

   Incorrect Sentence:

   Bangla: *theke gaachha phala pa.De*
   English: *from tree fruit falls.*

   Here the Post position *theke (from)* is placed before noun *gaachha (tree).*

   Correct Sentence:

   Bangla: *gaachha theke phala pa.De.*
   English: *Fruit falls from tree.*

2. Addition Operations:
   a) Repeated words:
      Bangla: *aami ekati *bhaala bhaala Chele*
      English: *I am a *good good boy*
   b) Unnecessary words:
      Bangla: *paramaaNu anu apekShaa *adhika kShudratara*
      English: *atom is *more smaller than molecule.*
3. Deletion Operations:
   a) Implicit Subject:
      Bangla: **[ ] tomaara maŇgala karuna* (Subject `iishbara` is missing here)
      English: *May *[ ] bless you.* (Subject: *God* is missing here)
   b) Implicit Verb:
      Bangla: *tumi ki maadhyamika pariikShaa *[ ]* (Verb: *debe* is missing here)
      English: *Will you *[ ] matriculation exam?* (Verb: *give* is missing here)
4. Substitution Operations:
   a) Similar word or Cohort replacement:

      Incorrect Sentence:

      Bangla: **bale baagha thaake*[1]
      English: **tell tiger lives*

[1] All Bangla examples are given in ITRANS format.

* Indicates error word in the sentence.

Correct Sentence:
Bangla: *bane baagha thaake*
English: *Tiger lives in forest*

Here *bale* (tell) and *bane* (forest) are cohorts of each other but *bale* is verb and *bane* is noun. In literature this type of error is also known as real word spelling error.

Types of Grammatical Errors

1. Tense Error:
   Example 1:
   Bangla: *aami prashnapatra pa.Daba o uttara diYechhilaama.*
   English: *I will read the question paper and I gave the answer.*
   Example 2:
   Bangla: *gatakaala aami sinemaa Jaaba*
   English: *Yesterday I will go to Cinema.*
   Example 3:
   Bangla: *Jakhaana aami darajaa khulachhilaama takhana se ghare Dhuke pa.Dechhila*
   English: *When I was opening the door then he entered the room.*
2. Person Error:
   Example:
   Bangla: *chhaatraraa nishchaYa bidyaalaYa Jaabe Jadi *se pariikShaa dite chaaYa.*
   English: *student must goes to school if *he wants to appear in the exam.*
   Plural sense of student has been lost by the singular representation of 'he'.
3. Case Error: case marker associated with pronoun and noun may be replaced. For example in the sentence *eTaa *kaakaaraa ba_i (English: This is uncle's book)* the suffix *raa* of the noun *kaakaa (uncle)* is changed from genitive case '*ra'*.
4. Adjectival Suffix Error: In the sentence **daYaamaYii shikShaka aasachhena (English: The kind-hearted teacher is coming)* the female suffix *maYii* of the word *daYaa (kindness)* is changed from male suffix *maYa* which goes with *shikShaka (male teacher).*
5. Improper use of punctuation:
   Example 1:
   Bangla: *tomaara naama ki* ।
   English: *What is your name.*
   Here the punctuation । is used instead of '?' symbol.
   Example 2:
   Bangla: *aami*, dekhalama se aasachhe* ।
   English: *I, see he is coming.*

6. Sentence Fragment:
   Example:
   Bangla: *aami gaana gaa_iba *। jadi tumi naacha* ।
   English: *I will sing. if you dance.*
7. Invalid Subject-Verb agreement:
   Subject and Verb have to agree with respect to number and person. *aami bhaata *khaabena (English: I eat rice)* is an incorrect sentence because the subject *aami (I)* is the first person non honorific but the person information of the verb *khaabena (eat)* is third person honorific.
8. Count Error:
   Example:
   Bangla: *aamaara tinajana bandhu aachhe : jaYanta, raajiiba, debaaruna o saurabha* ।
   English: *I have three friends: Joyanta, Rajib, Debarun and Saurabh.*

2.2 Previous Work

Stemberger [4] introspects the performance error of native speaker spoken language and reports proportion of the four types of error as follows: substitution (48%) > insertion (24%) > deletion (17%) > combination (11%). Foster [3] has manually created an error corpus for English and has classified missing word errors based on Part of Speech tag of this missing word. According to her "98% of the missing parts-of-speech come from the following list (the frequency distribution in the error corpus is given in brackets): det (28%) >verb (23%) > prep (21%) > pro (10%) > noun (7%) > to (7%) > conj (2%)". But manually creation of such corpus is very time consuming and non trivial task. Brockett et al. [15] created an artificial error corpus by introducing mass/count noun errors. They treated the error correction task in the machine translation point of view. Their aim was to apply Statistical Machine Translation (SMT) technique for converting ungrammatical sentences containing mass/count noun errors to grammatical sentences. Wagner, Foster, and Genabith [2] have suggested a novel approach of automated error corpus creation. They have carried out a detailed analysis of Missing Word Errors, Extra Word Errors, Agreement Errors and Covert Errors. Lee and Seneff [14] created artificial error corpora by introducing verb form errors. To mimic the real life errors, Foster and Anderson [16] designed the GenERRate tool. Their algorithm generates error corpus by introducing error along the line of the previously specified real life error templates.

3 Experimental Data Set

For our analysis, Bangla well-formed unicode sentences were collected from the web of various domains including literature, science, sports, music and news wire (2005-2010). We assumed that the syntax and semantics of the collected sentences are correct as they are mostly collected from different news wires which are normally edited and proof-read. Corpora from multiple domains have been collected to avoid

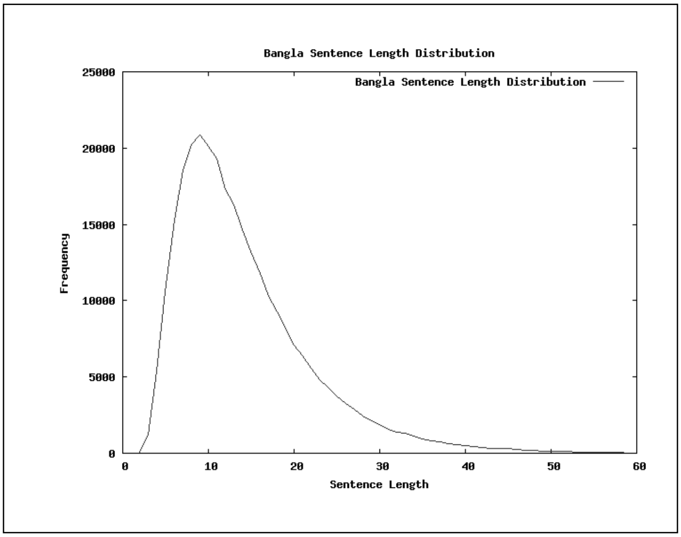


Fig. 2. Bangla Sentence Length Distribution

the skewed distribution of data. From this set of collected Bangla sentences (approx 4 lakh 80 thousand), sentence length distribution has been measured. It is found that sentences containing 11 words are the most frequent in this corpus. Figure 2 shows the Bangla Sentence length distribution.

4 Methodology

Now we will discuss our novel approach for error corpus generation. The procedure is as follows:

Step-1

If a grammatical sentence contains n words then transposition between two consecutive words can generate (n-1) sentences with assumption that only one transposition done in each sentence. Table 1 shows 3 sentences generated from a sentence containing 4 words. Though the last two examples in the table are grammatically correct, but transposition-2 is semantically weird and transposition-3 is relatively uncommon.

Table 1. Examples of Transposition Operation

Operation	Example
Source	*gaachha theke phala pa.De*[2]
Transposition -1	*theke gaachha phala pa.De*
Transposition -2	*gaachha phala theke pa.De*
Transposition -3	*gaachha theke pa.De phala*

Step-2

Transposition of highly collocated sequences surely induces noise in a grammatical sentence. Erroneous sentences have been automatically generated by changing the word order of different types of Bangla collocated words sequences collected from the corpus. We distinguish between the following three categories: echo words (if w_1w_2 is a word sequence and w_2 has no meaning), hyphenated words (w_1 and w_2 are connected by hyphen) and highly collocated words. Extraction of echo words and hyphenated words is simple. One can use a simple regular expression [a-zA-Z]+ \-[a-zA-Z]+ for collecting hyphenated words from corpus and [\s\a]([a-z]([a-z]+)\s+[a-z]\2)[\s\a][3] for collecting echo words. For collecting collocated and co-occured word sequences from corpus, a statistical approach [17] has been used. Variance(σ^2) of the number of words separating word w_2 from word w_1 have been estimated and low variance word sequences have been filtered using a statistical significance test (t-test) with 99.5% confidence level. The null hypothesis H_0 is that the word sequences (w_1w_2) appear independently in the corpus. These filtered word sequences are cross verified with Mutual Information (MI) values between w_i and w_j. The word sequences having higher Mutual Information and lower variances and having t-value greater than 2.57 (considering α = 0.005) have been considered as collocated words. MI between words w_1 and w_2 has been estimated as follows:

$$MI(w_1, w_2) = \log_2 \frac{p(w_1, w_2)}{p(w_1).p(w_2)} \tag{1}$$

where $p(w_1, w_2) = \dfrac{Count(w_1, w_2)}{N}$

and $Count(w_1, w_2)$ is the number of sentences in which w_1 and w_2 co-occur and N is the number of sentences in the training corpus . Accordingly the probability of the denominator of Equation (1) is calculated.

[2]

Bangla Sentence:	*gaachha*	*theke*	*phala*	*pa.De*
English Word Meaning:	Tree	from	fruit	fall
English Translation:	Fruit falls from tree			

[3] Python regex notation has been used here.

Step-3

Another way of generating erroneous sentences is by replacing a word with its cohorts and homophones. Cohorts are generated using regular expression by adding, deleting or substituting a single character or moving character sequences in a word. These generated words are then verified with spelling dictionary to ensure that the generated words are correctly spelled. In this process, if we assume that k number of words/cohorts can be generated on an average from a single word then k x n sentences can be generated from a sentence containing n words. Instead of k^n sentences, k x n sentences are generated as we are considering just replacement of one word at a time. We can reduce the value of k by considering only the nearest neighbor 4 keys (UP, DOWN, LEFT, and RIGHT) of the keyboard position for a particular character of a word in the time of generating cohort. Levenshtein Distance [18] (Edit Distance) also can be used to prune the over generated cohort words. Words having minimum edit distance with the original word are selected for the cohort list.

Step-4

By deleting a particular word from a sentence containing n words we can generate n sentences where each sentences containing (n-1) words. Table 2 shows 4 sentences generated from a sentence containing 4 words where each sentence containing 3 words.

Table 2. Examples of Deletion Operation

Operation	Example
Source	*gaachha theke phala pa.De*
Deletion - 1	*theke phala pa.De*
Deletion - 2	*gaachha phala pa.De*
Deletion - 3	*gaachha theke pa.De*
Deletion- 4	*gaachha theke phala*

Step-5

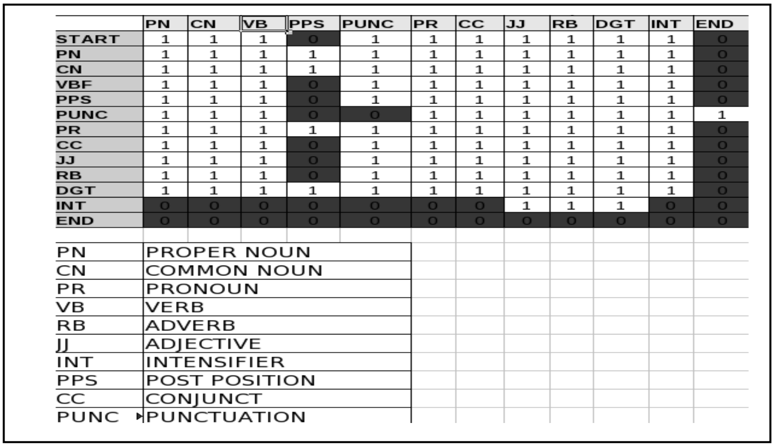
By addition a word from a vector $\vec{W} = \begin{bmatrix} w_1 \\ w_2 \\ w_3 \\ \vdots \\ w_v \end{bmatrix}$ in (n+1) possible position of a sentence containing n words, we can generate V x (n+1) sentences where V is the length of the vector. Here we are considering one word is inserted at a time. Table 3 shows number of sentences generated by addition operation. Thus applying step-1 to step-5 we can generate approximately (n-1)+ k x n+ n + V x (n+1) sentences from a sentence containing n words.

Table 3. Examples of Addition Operation

Operation	Example
Source	*gaachha theke phala pa.De*
Addition 1	$\vec{W}$ *gaachha theke phala pa.De*
Addition 2	*gaachha* $\vec{W}$ *theke phala pa.De*
Addition 3	*gaachha theke* $\vec{W}$ *phala pa.De*
Addition 4	*gaachha theke phala* $\vec{W}$ *pa.De*
Addition 5	*gaachha theke phala pa.De* $\vec{W}$

Step-6

Figure 3 shows a N x N tag association matrix which is generated after analyzing 5000 manually parts-of-speech (POS) tagged Bangla sentences having different syntactic categories. Every possible combination of two POS tag sequence is searched programmatically from this tagged corpus. On successful match, each cell of the matrix corresponding to the tag sequence is filled with 1, otherwise the cell contains 0. The cell with zero value indicates an invalid relationship i.e. POS tag of column N_i can not occur after tag of row N_j. In other words POS tag of N_i does not follow tag N_j row. For example Post position (PPS) cannot appear after intensifier (INT). Consulting this matrix, mal-rule can be generated which can be used for transposition of the word sequence of a sentence after being annotated by an automatic POS tagger.



	PN	CN	VB	PPS	PUNC	PR	CC	JJ	RB	DGT	INT	END
START	1	1	1	0	1	1	1	1	1	1	1	0
PN	1	1	1	1	1	1	1	1	1	1	1	0
CN	1	1	1	1	1	1	1	1	1	1	1	0
VBF	1	1	1	0	1	1	1	1	1	1	1	0
PPS	1	1	1	0	1	1	1	1	1	1	1	0
PUNC	1	1	1	0	0	1	1	1	1	1	1	1
PR	1	1	1	1	1	1	1	1	1	1	1	0
CC	1	1	1	0	1	1	1	1	1	1	1	0
JJ	1	1	1	0	1	1	1	1	1	1	1	0
RB	1	1	1	0	1	1	1	1	1	1	1	0
DGT	1	1	1	1	1	1	1	1	1	1	1	0
INT	0	0	0	0	0	0	0	1	1	1	0	0
END	0	0	0	0	0	0	0	0	0	0	0	0

PN	PROPER NOUN
CN	COMMON NOUN
PR	PRONOUN
VB	VERB
RB	ADVERB
JJ	ADJECTIVE
INT	INTENSIFIER
PPS	POST POSITION
CC	CONJUNCT
PUNC	PUNCTUATION

Fig. 3. POS tag association matrix

4.1 Confidence Score and Mal-rule Filters

Following the above mention procedure, we can generate erroneous sentences from a corpus of grammatical sentences. Our procedure generates approximately *{(n-1)*+ *k* x *n* + *n* + *V* x *(n+1)* } sentences from a sentence containing n words. Therefore, the number of generated sentences using this method increases with the number of words in a grammatical sentence. We have seen that the mode of the sentence length distribution of our collected Bangla corpora is 11. This implies that the upper bound of the number of sentences generated by our procedure is *10*+ *k* x 10 + *10* + *V* x 11. Those many sentences can be generated from a single sentence having 11 words. If we have 22000 11-word sentences in our corpus of approximately 480000 grammatical sentences, then 22000 * {*10*+ *k* x *10* + *10* + *V* x *11*} sentences can be generated using our method. Some Bangla sentences may have as many as 57 words but we are not considering such cases as such sentences are very infrequent (See Figure 2). Therefore filtering ungrammatical sentences from this set of *{(n-1)*+ *k* x *n* + *n* + *V* x *(n+1)* } sentences is not a trivial task. In this stage proper sampling is required so that sentences indicative of more frequently made errors have higher probability of getting selected. Therefore we have applied both rule-based and statistical based approach for collecting significant sample from this population. Initially we pass the sentences though our HMM based semi-supervised POS tagger and then generated tag sequences are pass through mal-rule detector which collect the sentences containing improper pos tag sequences. We also have calculated the confidence score of each sentence by calculating bigram, Mutual Information (MI) and Relative Position Score [10]. A numeric score is assigned to determine the quality of the sentence. The sentence-level confidence measure is based on the score of each and every individual word in the sentence. Confidence score estimation using N-gram, measures the grammatical soundness of the sentence and MI based confidence score, measures the lexical consistency [19]. MI is used to detect presence of which word reduces the uncertainty of appearance of another word in the same sentence. Confidence score of a sentence using MI has been calculated as follows:

$$
\begin{aligned}
Sore(S) &= Score(w_1, w_2, w_3 \cdots w_n) \\
&= \frac{1}{n}\sum_{i=1}^{n} Score(w_i) = \frac{1}{n}\sum_{i=1}^{n} \frac{\sum_{j=1, j\neq i}^{n} MI(w_j, w_i)}{n-1}
\end{aligned}
\tag{2}
$$

Here $MI(w_j, w_i)$ is calculated using equation (1). MI based confidence measure do not take word order into account. It focuses on long range lexical relationships. For this reason, we have also estimated the relative position based confidence score. Confidence score of a sentence using Relative Position Score [10] has been calculated as follows:

$$RP_{Score}(S) = RP_{Score}(w_1, w_2, w_3 \cdots w_n) = \frac{\displaystyle\sum_{j=2}^{n}\sum_{i=1}^{j-1} \frac{\left(\dfrac{freq_{Dep}(w_i, w_j)}{freq_{Ind}(w_i, w_j)}\right)}{j-1}}{n-1} \tag{3}$$

where $freq_{Dep}(w_i, w_j)$ is the number of sentences in which w_i and w_j co-occur with a constraint that w_j appear after w_i in a sentence and $freq_{Ind}(w_i, w_j)$ is the number of sentences in which w_i and w_j co-occur without any positional constraint . Mutual Information has been used for proper selection of the erroneous sentences generated by substitute operation. Low Mutual Information ensures that a word in the sentence is wrongly placed in the context of the other words. Bigram and Relative position scores have been used to select the erroneous sentences generated by transposition operations. The error corpora creation procedure with an English example is shown in Figure 4.

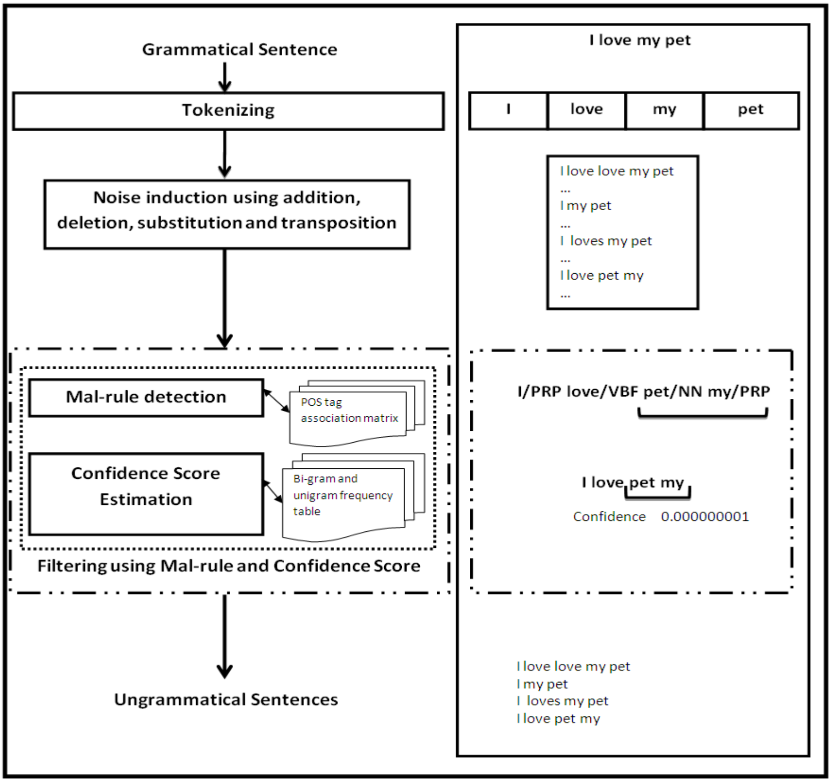
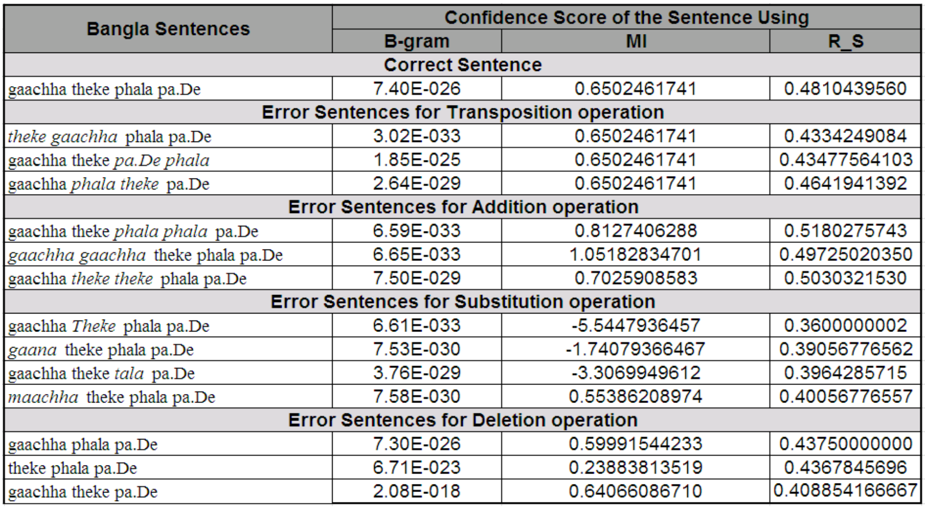


Fig. 4. Simplified functional diagram of automatic error corpora creation

5 Result and Discussion

Following the experimental procedure described in Section 4 we have generated erroneous sentences from randomly selected 1000 sentences from a corpus of grammatical sentences. Then these generated ill-formed sentences are filtered using mal-rule detector and depending on the confidence score (see sub section 4.1). After manually analysing the random sample of generated ill-formed sentences, we found that 87% of generated sentences are really ungrammatical. Most of these generated sentences have invalid POS tag sequences. Though some of the generated sentences have valid POS tag sequences but the word sequences in these sentences are infrequent. Experimental result also shows that 13% of that generated sentences are grammatical because insertion, deletion and substitution operation some time generates another grammatical construction. Figure 5 shows sample of Bangla erroneous sentences generated by our method from a grammatical sentence with their aforementioned confidence score. In this figure, the first sentence is a correct sentence and the remaining erroneous sentences are generated automatically. In this figure R_S indicate the relative position score of a sentence.



Bangla Sentences	Confidence Score of the Sentence Using		
	B-gram	MI	R_S
Correct Sentence			
gaachha theke phala pa.De	7.40E-026	0.6502461741	0.4810439560
Error Sentences for Transposition operation			
theke gaachha phala pa.De	3.02E-033	0.6502461741	0.4334249084
gaachha theke *pa.De phala*	1.85E-025	0.6502461741	0.43477564103
gaachha *phala theke* pa.De	2.64E-029	0.6502461741	0.4641941392
Error Sentences for Addition operation			
gaachha theke *phala phala* pa.De	6.59E-033	0.8127406288	0.5180275743
gaachha gaachha theke phala pa.De	6.65E-033	1.05182834701	0.49725020350
gaachha *theke theke* phala pa.De	7.50E-029	0.7025908583	0.5030321530
Error Sentences for Substitution operation			
gaachha *Theke* phala pa.De	6.61E-033	-5.5447936457	0.3600000002
gaana theke phala pa.De	7.53E-030	-1.74079366467	0.39056776562
gaachha theke *tala* pa.De	3.76E-029	-3.3069949612	0.3964285715
maachha theke phala pa.De	7.58E-030	0.55386208974	0.40056776557
Error Sentences for Deletion operation			
gaachha phala pa.De	7.30E-026	0.59991544233	0.43750000000
theke phala pa.De	6.71E-023	0.23883813519	0.4367845696
gaachha theke pa.De	2.08E-018	0.64066086710	0.408854166667

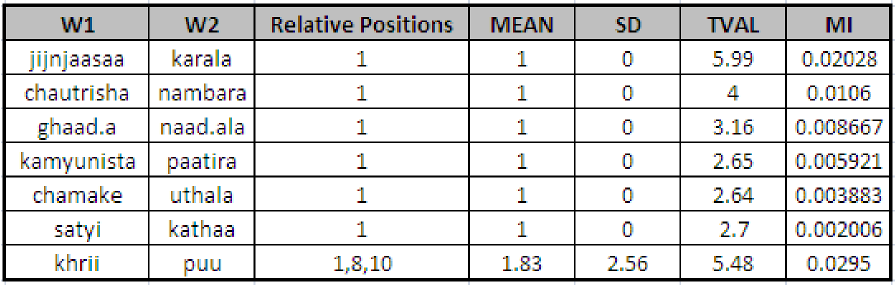
Fig. 5. Erroneous sentences generated from a single sentence and selected according to the confidence score

Using echo words, hyphenated words and collocation collection methodology as discussed in the step 2 of section 4, we have collected desired results. Table 4 shows Bangla Echo words and Hyphenated words collected from the corpus.

Table 4. Bangla Echo words and Hyphenated words

Echo Words	Hyphenated Words
oShudha TaShudha	*aNu-paramaaNu*
kha_i Ta_i	*adala-badal*
goYendaa ToYendaa	*anumata-abhimata*
chakaara bakaara	*asukha-bisukha*
chaNDaala phaNDaala	*aaina-aadaalata*
jaata paata	*kaapa.Da-chopa.Da*
nardamaa Tardamaa	*Kaamanaa-baasanaa*

Transposition between them might cause error to be induced in a sentence. Transpositions of echo words are not allowable but transpositions of hyphenated words are allowed sometime. For example we may sometimes use "*baasanaa-Kaamanaa"* in place of *"Kaamanaa-baasanaa"*, though these appearances are very infrequent. Figure 6 shows some automatically collected collocated and co-occured word sequences along with their relative position, mean and variance of relative positions, t-value and Mutual Information between these word sequences. Transposition of automatically collected echo words, hyphenated words and collocated words induce noise in a grammatical sentence and this procedure of automatic induction of noise gives a very good result.



W1	W2	Relative Positions	MEAN	SD	TVAL	MI
jijnjaasaa	karala	1	1	0	5.99	0.02028
chautrisha	nambara	1	1	0	4	0.0106
ghaad.a	naad.ala	1	1	0	3.16	0.008667
kamyunista	paatira	1	1	0	2.65	0.005921
chamake	uthala	1	1	0	2.64	0.003883
satyi	kathaa	1	1	0	2.7	0.002006
khrii	puu	1,8,10	1.83	2.56	5.48	0.0295

Fig. 6. Erroneous sentences generated from a single sentence and selected according to the confidence score

6 Conclusion

In this paper, we discussed practical issues pertaining to automatically creating an error corpus by combining statistical and linguistic knowledge. Types of errors in the time of writing text are analysed in detail. Then a methodology of automatic error corpus creation with appropriate manual intervention has been discussed. Issues pertaining to creating erroneous sentences resulting from pronoun referencing error, state error, time error, and other semantic errors fall outside the scope of this paper. Though the present work focuses on the most frequent grammatical errors in Bangla written text, detail taxonomy of grammatical errors in Bangla is also presented here, with an aim to increase the coverage of the error corpus in future.

As part of future work, we plan to devise a more principled approach to sampling the auto generated error corpus in the boundary cases and also to ensure that automatically generated error sentences will mimic the naturally occurring learners' errors. A statistical classifier can make use of active learning to bootstrap the corpus creation process. We hope that the research reported in this paper encourages other researchers in Indian Languages to build robust grammar checkers using the error corpus we built and also contribute further to the growth of the corpus. A similar approach combining linguistic and statistical approach can also be tried for developing error corpora in other Indian Languages where such resources are not available as of now.

References

1. Kamp, H., Reyle, U.: From Discourse to Logic:Introduction to Modeltheoretic Semantics of Natural Language, Formal Logic and Discourse Representatio. Studies in Linguistics and Philosophy. Kluwer Academic Publishers (1993)
2. Wagner, J., Foster, J., van Genabith, J.: A Comparative Evaluation of Deep and Shallow Approach to the Automatic Detection of Common Grammatical Error. In: Proceedings of the Joint Conference on Empirical Methods in Natural Language Processing and Computational Natural Language Processing, pp. 112–121 (2007)
3. Foster, J.: Good Reasons for Noting Bad Grammar: Empirical Investigations into the Parsing of Ungrammatical Written English, Phd. Thesis, University of Dublin, Trinity College, Dublin, Ireland (2005)
4. Stemberger: Syntactic errors in speech. Journal of Psycholinguistic Research, 313–345 (1982)
5. Thurmair, G.: Parsing for Grammar and Style Checking. In: Proceedings of the 13th International Conference on Computational Linguistics, pp. 365–370 (1990)
6. Bustamante, F.R., Leon, F.S.: GramCheck: A grammar and style checker. In: Proceedings of COLING, pp. 175–181 (1996)
7. Stanley, Goodman: An empirical study of smoothing techniques for language modeling. In: Proceedings of the 34th Annual Meeting on Association for Computational Linguistics (1996)
8. Dagan, I., Karov, Y., Roth, D.: Mistake-Driven Learning in Text Categorization. In: The Second Conference on Empirical Methods in Natural Language Processing, pp. 55–63 (1997)
9. Powers, D.M.W.: Learning and Application of Differential Grammars. In: Proceedings Meeting of the ACL Special Interest Group in Natural Language Learning, pp. 88–96 (1996)
10. Liu, C., Wu, C., Harris, M.: Word Order Correction for Language Transfer Using Relative Position Language Modeling. In: Proceedings of 6th ISCSLP, pp. 1–4 (2008)
11. Michaud, L.N., Mccoy, K.F.: An intelligent tutoring system for deaf learners of written English. In: Proceedings of the Fourth International ACM SIGCAPH Conference on Assistive Technologies, pp. 13–15

12. Leacock, Chodorow, Gamon, Tetreault: Automated Grammatical Error Detection for Language Learners. Morgan & Claypool Publishers (2010)
13. Sjobergh, Knutsson: Faking errors to avoid making errors: Very weakly supervised learning for error detection in writing. In: Proceeding of the International Conference on Recent Advances in Natural Language Processing, pp. 506–512 (2005)
14. Lee, Seneff: Correcting misuse of verb forms. In: Proceeding of the 46th Annual Meeting of the Association for Computational Linguistics: Human Language Technology, pp. 174–182 (2008)
15. Brockett, Dolan, Gamon: Correcting ESL errors using phrasal SMT techniques. In: Proceedings of the 21st International Conference on Computational Linguistics and 44th Annual Meeting of the Association for Computational Linguistics, pp. 249–256 (2006)
16. Foster, Andersen: GenERRate: Generating errors for use in grammatical error detection. In: Proceedings of the Fourth Workshop on Building Educational Applications Using NLP, pp. 82–90 (2009)
17. Manning, C.D., Schütze, H.: Foundations of Statistical Natural Language Processing. MIT Press, Cambridge (1999)
18. Levenshtein, V.I.: Binary codes capable of correcting deletions, insertions, and reversals. Soviet Physics Doklady 10, 707–710 (1966)
19. Raybaud, S., Langlois, D., Smaïli, K.: Efficient combination of confidence measures for machine translation. In: Proc. INTERSPEECH, pp. 424–427 (2009)