Tiny Corpus Applications with Transformation-Based Error-Driven Learning: Evaluations of Automatic Grammar Induction and Partial Parsing of SaiSiyat

Zhemin LIN
Graduate Institute of Linguistics,
National Taiwan University
1, Roosevelt road sec. 4
Taipei, R.O.C.
lin.zhemin@gmail.com

Li-May SUNG
Graduate Institute of Linguistics,
National Taiwan University
1, Roosevelt road sec. 4
Taipei, R.O.C.
limay@ntu.edu.tw

Abstract

This paper reports a preliminary result on automatic grammar induction based on the framework of Brill and Markus (1992) and binary-branching syntactic parsing of Esperanto and SaiSiyat (a Formosan language). Automatic grammar induction requires large corpus and is found implausible to process endangered minor languages. Syntactic parsing, on the contrary, needs merely tiny corpus and works along with corpora segmented by intonation-unit which results in high accuracy.

1 Introduction

SaiSiyat is a Formosan Austronesian language with less than 4,677 speakers (1995 census data). It is an SOV language with four verbal voices, six case markers, but without declensions (Yeh (2000)). As other Austronesian languages in Taiwan, SaiSiyat writing system is just officially standardised. [1] Few written materials are published in this language and the main source of its corpora is linguistic fieldwork in form of transcription of oral narration and conversation. The tiny scale of corpora makes it hard to do probabilistic natural language processing. Other affordable methods to build a syntactically tagged treebank are thus subjects to our work.

SaiSiyat parallels to ancient Egyptian in terms of the description of Rosmorduc (published on Internet). Part of its grammar is still unsure. Grammatical errors are found in texts. The absence of punctuation makes the corpus impossible to be proceeded at sentence level. In order to partially parse this language, the applications of Kullback-Leibler divergence and transformation-based error-driven learning (TBL) are evaluated in the paper.

NTU SaiSiyat corpus (**?**))contains 27 texts, 3702 intonation units (IUs), 12065 words. Its notation follows the convention of Du Bois (1993). Sixteen narrations are composed in the corpus, including 4 Pear Stories (a colour mute film), 8 Frog Stories (a sketchbook by Mayer (1980)) and 4 indigenous legends. The corpus is tagged with a TBL tagger in reduced Penn Treebank Tagset. The overall accuracy is 88.11%. (Lin (2004)) Additional collected texts are added in our experiment to enlarge the corpus. An example of original and tagged data segment follows:

```
1. kor-koring     min-a'rem     korkoring/NN mina'rem/VB
   Red-discipline MIN-rest
   "A child was asleep."
```

Esperanto is planned as an international help language in 1887 by L. Zamenhof. [2] Large corpora of authentic journals, translated works and archives of Yahoo!Groups are available online for free. Its declension and conjugation are regular, permitting us to tag the texts easily, quickly and correctly. We

[1] A standardised spelling system of Formosan Austronesian languages is published by the Council of Indigenous Peoples, Executive Yuan. Diversity still exists among their users.

[2] Cf. http://eo.wikipedia.org/wiki/Esperanto

choose "Monato" archive, a periodic written in the language, as a sentence-based contrast. See table 1 for corpora statistics.

Table 1: Corpora data

Language	Size	Words	Vocab.	Tags	Sentence Length
SaiSiyat	3,888 IU	13,970	1,697	20	4.15
Esperanto	84,496 sent.	1,556,566	125,663	27	18.49

2 Phrase Structure Grammar with K-L Divergence

Kullback-Leibler divergence as reported in Brill and Markus (1992) measures the distributional similarity of two sets of tags. For each set of tags in similar environment, a binary-branching rule $tag_x \rightarrow tag_y tag_z$ is built and their similarity measured. A context-free grammar is hence reduced to finding the nearest path to collapse a sentence (or a segment of words, in case of SaiSiyat) into a single tag.

The relative entropy (1) of one set of tags is first calculated to estimate the amount of extra information necessary to describe another set. The divergence (2) between two sets is the sum of the amount of necessary data for describing each other, serving as a measure of the difference of their distributions.

$$D(P_1||P_2) = \sum_{x \in Env} P_1(x) * \log \frac{P_1(x)}{P_2(x)} \tag{1}$$

$$D_{1,2} = D(P_1||P_2) + D(P_2||P_1) \tag{2}$$

The "environment" may be surrounding words or tags. For example, *This is John/NNP .* and *This is a/DT chair/NN .* , we find **NNP** and **DT-NN** occurring between *"is"* and *"."*. The environment is schematically written as **word _ word**. However, we may not have enough environments in case of a tiny corpus. *word _ word* can be replaced by *tag _ tag*, but lexical information is discarded if we made this change. As a result, a transitive verb is confounded with an intransitive verb with a preposition (*VBD → VBD IN* in Brill & Marcus' example) for their high frequency in the same environment. They try to multiply the divergence with the mutual information (3) of tag_y and tag_z in order to exclude grammatically unbounded set of tags, such as **VBD IN** (4).

$$H(tag_i\_) = - \sum_{tag_j \in Tagset} p(tag_j|tag_i) * log_2 p(tag_j|tag_i) \tag{3}$$

$$D(P_1||P_2) * (H(tag_y, tag_z, \_) - H(tag_y, \_)) \tag{4}$$

The example is found in our experiment, showing that the rule of pronoun → noun phrase (*PRP → DT NN*) gets adjusted to a lower (better) score,

```
Divergence: PRP DT NN 1.06991078978
Adjusted:   PRP DT NN 0.183881076959
```

Since the system may find a large sum of rules, only the 15 best scored rules of each possible set of $tag_y tag_z$ are applied in path finding. For each path which permits to reduce a string into a single tag, the sum of divergences along the path is calculated. The path with the lowest (thus the best) score is considered the correct one. For example, *korkoring/NN min'itol/VB ila/ASP* (lit. "child rest PERFECT_MARKER") may be reduced by the following rules:

```
   (begin state)              NN VB ASP
1  VB VB ASP 0.0864877817911  NN VB
2  NN NN VB  0.235779052654   NN
   (end)
```

The path is *NN VB ASP* → *NN VB* → *NN*, scored (1) + (2) = 0.32226683444510001.
We adopt the Brill & Marcus model, and filtered each corpus by the following criteria:

- No UNK (unknown) tag.
- Sentence length: 2 to 20 words.
- The correctness is judged by maximum match to government-binding theory.

In case of *AT JJ NN*, for example, the one bracketed as `(AT (JJ NN))` is considered correct.

2.1 Esperanto Corpus

Since there are more than 1 million words in the Esperanto corpus, we apply the *word* _ *word* schema in order to get a more precise measurement. 10,384 rules are generated at the first stage. After the 15 best rules are chosen, 6,980 rules remain. The method is found to require a huge search space, consuming far more computation time than we could afford. Therefore only sentences with 3 to 5 words are reported for their correctness (see table 2).

Table 2: Result of K-L divergence (Esperanto)

Sentence Length	Offset	Search Space
3	15	379
4	146	> 5000
5	3365	> 5000

Average offset and average search space of 10 sentences of each length are reported in the table. There is a good reason not to give each of them a score. Since merely the path with lowest/best score are considered right, and we have no external data to decide if some higher scored rule should be the right one, we can just demonstrate the distance between our result and the ideal. Among 30 test sentences, only 1 sentence is parsed correct.

Below is an example of one test sentence of each length, the offset of the correct path, the search space, path score and how they are reduced into one tag. Non-terminal labels are ignored in the model.

```
WRB VBP NN          [   24/ 357] 0.0548143199314 .(WRB,RP(VBP,NN))
WP NN IN NN         [  112/5000] 0.0669057743547 #(WP,RP(NN,RB(IN,NN)))
NNP VB JJ JJ NNS    [ 2513/5000] 0.0446834608909
                                      UH(NNP,WRB(VB,PRP$(JJ,.(JJ,NNS))))
```

2.2 SaiSiyat Corpus

Phenomena such as repair, repetition and recover occur frequently in oral data. The constituent of [spec,CP] (e.g. complementiser *that*) tends to stay at the final position of the main IU. Case marker (CM) and its marked noun (NN) are often separated in two conjoint IUs whenever the speaker needs time to recall a word. However, a IU-based corpus seems to provide more information to the extent of frequent collocating constituents. For example, for the following input, we induce easily the right rule of *VB* → *VB ASP*:

korkoring/NN mina'rem/VB (lit. "Child sleeps.")
ahoe'/NN mwa:i'/VB ila/ASP (lit. "Dog came ASP.")

This observation is further proven by our experiment. 3,473 rules are first generated and 2,980 rules remain for finding paths. The correctness of 10 IUs of lengths 3, 4 and 5 are reported in table 3.

Table 3: Result of K-L divergence (SaiSiyat)

Sentence Length	Offset	Search Space
3	1	385
4	17	> 5000
5	55	> 5000

The result is surprisingly good. Among 30 test sentences, 13 are correct (i.e. with the lowest score). It is observed, however, the result declines quickly as IU length increases.

2.3 Discussion

The correct path to parse a sentence is not always found to be the lowest scored one. We observed two major problems preventing this model to be useful for our task. First, some constituents tend to conjoin firmly, causing correct path to be scored either the best or far from the best. Below are first 5 paths in parsing `(NN (VB (VB ASP)))`:

```
#1  0.107512915137  ,(NN,,(UNK(VB,VB),ASP))
#2  0.108052449446  .(NN,.(UNK(VB,VB),ASP))
#3  0.111216713424  NNP(NN,.(UNK(VB,VB),ASP))
#4  0.111984096398  NNP(UNK(EX(NN,VB),VB),ASP)
#5  0.112173131487  .(NN,.(VB,PRP$(VB,ASP)))
```

It is clear that the low divergence of `(VB VB)` causes other paths incompetent. Second, the huge search space enforces Brill & Marcus to implement the *beam search*. Without implementing this, we suffer for the computation time. The search space is calculated in the following formula:

$$space = t^{(n-1)} * f(n)$$
$$f(n) = 2, n = 2$$
$$f(n) = 5, n = 3$$
$$f(n) = \sum_{i=3}^{n-1} \frac{i*(i+1)}{2} - (n-3) = \frac{n^3-7n-6}{6}, n > 3$$

Let n be sentence length, t be the average amount of the left value of each right hand rule (in our case, 15), the possible paths for a 15-word sentence is approximately $1.6 * 10^19$. We once tried to parse a simple sentence like

> "Alie vi povas serĉi nur en la Antaŭparolo kaj Ekzercaro de la Fundamento de Esperanto."

and the computer hanged. This is definitely uncomputable.

3 Syntactic Tree Acquisition

Brill (1993) offers another way to produce parsing tree. Each sentence is parsed in a naive manner before being fed into a learner. The learner holds the result of a iterative learning process of comparing the input tree and the golden corpus ("truth"). The highest scored rule in each iteration is acquired. The TBL model is shown in figure 1.

For example, the sentence

```
ray babaw hayza' ka 'ilaS.   ray/IN babaw/RB hayza'/EX ka/CM 'ilaS/NN
Loc above Ex     Nom moon
```

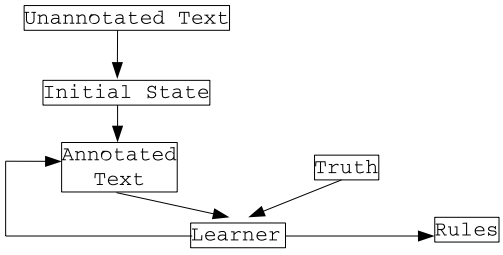

Figure 1: Transformation-based error-driven learning (adopted from Brill (1992))

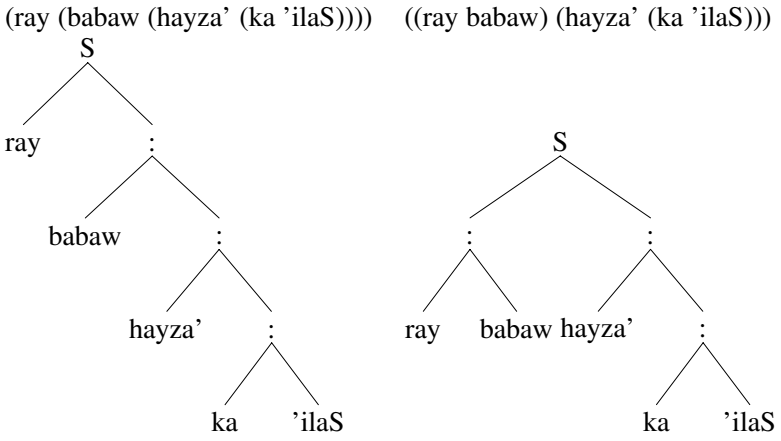

Figure 2: Tree mutation after correct transformation

is first right-bracketed as `(ray (babaw (hayza' (ka 'ilaS))))` and then transformed into `((ray babaw) (hayza' (ka 'ilaS)))`, resulting in the mutation of syntactic trees as shown in figure 2.

Twelve template rules are generated for each tag:

- *(Add—delete)* a *(left—right)* parenthesis to the *(left—right)* of POS tag X
- *(Add—delete)* a *(left—right)* parenthesis between tag X and Y

For example, the rule of *-LL NN* (delete left parenthesis to the left of NN) works as,

```
(   The ( dog   barked ) )
(   The   dog   barked ) )
(   The   dog   barked   )
( ( The   dog   barked ) )
( ( The   dog ) barked ) )
```

If a rule fails to apply, nothing happens to the input sentence. The rule is scored by the number of non-crossing constituents in comparison to the golden corpus, e.g.,

```
Test:    ( ( The big ) ( dog ate ) )
Golden:  ( ( The ( big dog ) ) ate )
```

There is one non-crossing constituent ("the") and 3 are crossing. The transformation is then scored 1 / 4.

Since we have to make a golden corpus of each language, 200 sentences/IUs are manually annotated. 150 sentences are randomly selected to train the learner and the remaining 50 sentences are randomly put in the test corpus.

3.1 Esperanto Corpus

The accuracy of the naive parser is 29.17%. The size of training corpus, the number of acquired rules and accuracy in terms of Brill's scoring system are shown in table 4, the error rate shown in table 5.

Table 4: Result of TBL parser (Esperanto)

# training	# rules	Accuracy
50	12	35.04%
100	24	25.64%
150	33	28.77%

Table 5: Error rate of TBL parser (Esperanto)

# training	0-error	$\leq$1-error	$\leq$2-error
50	32.0%	34.0%	36.0%
100	24.0%	24.0%	24.0%
150	26.0%	26.0%	26.0%

The accuracy is astonishingly low. In fact, we find the learner unable to seize good generalisation even inside the training corpus (see table 6).

Table 6: Accuracy of parsing a training corpus

# training	# rules	Accuracy
50	12	48.34%
100	24	49.86%
150	33	46.67%

This may be caused by the complexity of constituent composition in Esperanto grammar. We will return to this issue in §3.3.

3.2 SaiSiyat Corpus

Merely complete IUs are evaluated in the experiment since incomplete IUs do not form a close bracket. The naive accuracy of SaiSiyat corpus is 34.52%. The statistics is reported in table 7, the error rate in table 8.

The result is good (about 68%) but not good enough. The accuracy should be at least 80% for really practical task. The learner over-generalise too easily, preventing accurate parsing of complete IUs.

3.3 Discussion

Esperanto written sentences are often long and complex. Its word order is more free then the English one. Even more, subject is often post-posed and object or adverb is often moved to [spec,CP] position. This implies that a reduced tagset may not be distinguishing enough to catch the language fact. The

Table 7: Result of TBL parser (SaiSiyat)

# training	# rules	Accuracy
50	16	70.64%
100	15	67.43%
150	17	69.27%

Table 8: Error rate of TBL parser (SaiSiyat)

# training	0-error	$\leq$1-error	$\leq$2-error
50	68.0%	68.0%	74.0%
100	62.0%	64.0%	70.0%
150	64.0%	66.0%	72.0%

quick over-generalisation in parsing SaiSiyat implies a larger tagset as well. Yet we are unable to refine SaiSiyat tags since its lack of declension. For examine this assumption, we can tag Esperanto corpus with complete Penn Treebank tagset and redo the process. In fact, additional tag NNA, NNAS, JJA, JJAS, PRPA, PRPA, PRPAS are implemented to reflect Esperanto declension. The result is shown in table 9 and 10.

Table 9: Result of larger tagset

# training	# Simple	S-Accu	# Compl	C-Accu
50	12	35.04	14	35.04%
100	24	25.64	16	30.20%
150	33	28.77	28	38.46%

The tables show that a larger tagset works better then the reduced one. However, a larger tagset implies even larger training corpus. A training corpus with 500 Esperanto sentences is likely to result in high accuracy. This would be a dilemma if our purpose was to make the work faster and easier done.

4 Conclusion

The applications of automatic grammar induction using Kullback-Leibler divergence and syntactic parser based on transformation-based error-driven learning are evaluated in this paper. Since most Formosan Austronesian corpora contain less than 20,000 words, we have to deal with every possibility to process the languages with a computer in the critical task of language preservation. K-L divergence profits from a large corpus and is helpful only when a segment of text contains less than 3 words. This would not be very practical. Although a TBL parser is not as appealing as demonstrated in Brill (1993), the accuracy may be enhanced by a complex tagset or affordable (less than 1000) training corpus. It helps us at least to segment short phrases from continuous constituents and eases the work of building a human-polished treebank.[3]

[3]Chinese version of this paper and more information regarding this topic is accessible at `http://ljm.idv.tw/mywiki/DraftTinyCorpusApplication`. Esperanto tagger used here can be downloaded at `http://ljm.idv.tw/download/esptag-0.2.tar.gz`.

Table 10: Error rate of larger tagset compared with reduced tagset

# training	0-error		≤1-error		≤2-error	
50	36.0%	(+ 4%)	38.0%	(+ 4%)	38.0%	(+ 2%)
100	32.0%	(+ 8%)	32.0%	(+ 8%)	32.0%	(+ 8%)
150	36.0%	(+10%)	40.0%	(+14%)	40.0%	(+14%)

Acknowledgements

Our thanks go to SaiSiyat informants. We appreciate Michael Tanangkingsing for his brilliant fieldwork reports, Guido van Rossum for Python programming language and Linus Torvalds for Linux operating system. Michael Tanangkingsing also reviewed this article and checked the grammar.

References

Brill, E. and M. Markus. 1992. Automatically acquiring phrase structure using distributional analysis. In *DARPA Speech and Natural Language Workshop*, pages 155–159.

Brill, Eric. 1992. A simple rule-based part of speech tagger. In *Proceedings of ANLP-92, 3rd Conference on Applied Natural Language Processing*, pages 152–155, Trento, IT.

Brill, Eric. 1993. Automatic grammar induction and parsing free text: a transformation-based approach. In *Meeting of the ACL*, pages 259–265.

Du Bois, J. W., 1993. *Outline of Discourse Transcription*, pages 45–89. Hillsdale: Lawrence Erlbaum Associates, NJ.

Lin, Z. 2004. POS-tagger for SaiSiyat: using fieldwork notations and TBL. In *ROCLING XVI Student Workshop II*, pages 25–33.

Mayer, M. 1980. *Frog, Where are You?* Dial Books, NY.

Rosmorduc, S. published on Internet. Automata-guided context-free parsing for punctuationless languages. URL: http://citeseer.ist.psu.edu/363381.html.

Yeh, M. 2000. *Sàixiàyǔ Cānkǎu Yǔfǎ*. Yuǎnliú, Taipei.