Unification of Multiple Treebanks and Testing Them With Statistical Parser With Support of Large Corpus as a Lexical Resource

Dr. Ahmed Hussein Aliwy
Computer Science Department, University of Kufa/Kufa
Email:Ahmed_7425@yahoo.com

Received on:12/11/2015 & Accepted on:21/4/2016

ABSTRACT

There are many Treebanks, texts with the parse tree, available for the researcher in the field of Natural Language Processing (NLP). All these Treebanks are limited in size, and each one used private Context Free Grammar (CFG) production rules (private formalism) because its construction is time consuming and need to experts in the field of linguistics. These Treebanks, as we know, can be used for statistical parsing and machine translation tests and other fields in NLP applications. We propose, in this paper, to build large Treebank from multiple Treebanks for the same language. Also, we propose to use an annotated corpus as a lexical resource. Three English Treebanks are taken for our study which arePenn Treebank (PTB), GENIA Treebank (GTB) and British National Corpus (BNC). Brown corpus is used as a lexical resource which contains approximately one million tokens annotated with part of speech tags for each.

Our work start by the unification of POS tagsets of the three Treebank then the mapping process between Brown Corpus tagset and the unified tagset is done. This is done manually according to our experience in this field. Also, all the non-terminals in the CFG production are unified.All the three Treebanks and the Brown corpus are rebuilt according to the new modification.

Our test for the proposed unification are made in three types: (i) statistical parsing test for each Treebank alone without modification, (ii) statistical parsing test for each Treebank alone after the modification, (iii) statistical parsing test for the collection of the three Treebanks after modification ***without*** support of lexical resource, and (iv) statistical parsing test for the collection of the three Treebanks after modification ***with*** support of lexical resource. The unknown words are processed using a very simple suggested method.

We can show, simply in our work, that (a) the unification of multiple Treebanks can be done and will increase the accuracy. (b) A large annotated corpus as Brown corpus can be used for (i) decreasing the unknown words and (ii) we can extract the probabilities nearest to the reality. (c) The mapping between the unified tagset and the lexical tagset (used in Brown corpus) can be done straightforward.

Keywords: Treebanks unification, POS mapping, English Treebank, annotated corpus and Treebank.

INTRODUCTION

Parsing is one of the important tasks in natural language processing (NLP). It not an application but it is a stage in many NLP applications as machine translation (MT). Parsing is doing by an approach and grammar. There are many parsing approaches and little types of grammar. The most used grammar is Context Free Grammar (CFG) but, in most cases, according to the used application[1].

Some Parsing techniques can produce one tree and the others could produce all the acceptable trees for the given sentence. In general, the recognizer need to one tree only (any one not matter). In Machine Translation application, the parser should produce all trees and then select the best one. Producing all trees for a given sentence is the time waste task, therefore, some researchers introduced dynamic programing techniques in parsing as CYK, chart parsing and so on. Choosing the best tree is done by statistical methods by training from a Treebank. For

https://doi.org/10.30684/etj.34.5B.11
2412-0758/University of Technology-Iraq, Baghdad, Iraq

example producing Probabilistic Context Free Grammar (PCFG) from the Treebank where each rule in the grammar has probability according to its occurring in the Treebank. The tree which has the highest probability is taken as the best tree. Selecting best tree from many trees is also time consuming therefor sometimes Viterbi algorithm is used[2].

Most of the researchers took one Treebank or more and applied one approach or more on these treebanks in order to compare these approaches or the used treebanks. In this paper, we will take multiple Treebank. The rules of CFG and all the POS tagsets are unified for all these Treebanks which need to work manually. Brown corpus is taken as a lexical resource for classes of words in order to decrease the unknown words. This combination and unification of Treebanks with Brown corpus specify the novelty of our work. Our work can be summarized by the following steps:

1- Tagsets unification of the three Treebanks (It is done manually)

2- Unification of Non-terminals (in the CFG productions) of the three Treebanks (It is done manually)

3- The mapping between Brown Corpus tagset and the unified tagset.

4- Processing unknown words using a very simple suggested approach.

5- Training which was done in two separate methodologies:

a. Training and testing each Treebank separately before and after modification.

b. Training and testing the three Treebanks collectively after unification with and without using Brown corpus support.

Related work

There are many researches on probabilistic parsing using Treebank implemented for many world languages especially for the English language.

Sampo and Filip took two corpora from the biomedical domain as GENIA Treebank and BioInfer. TheirUnification processconvertedLink Grammar dependency scheme (LG) to the Stanford dependency scheme [1] with these two corpora.They, also,found that the performance of the BioLG parser is not adversely affected by the conversion [2].

Muhua and Jingbo unified multiple constituency treebanks. They converted annotations in one treebank to fit the standard of another Treebank where the test was done on two Chinese Treebanks: CTB and TCT[3].

Niu & Haifeng[4] did not use conversion rules but they proposed to use a parser to convert a Dependency Treebank (DTB) to a constituency Treebank. This was done by selecting conversion results from the best list produced by a parser for sentences in DTB when it is trained on a constituency Treebank.

Zhenghua, Ting and Wanxiang exploited multiple monolingual treebanks with different annotation guidelines for parsing. Several types of transformation patterns were used. They used (Penn Chinese Treebank 5.1 and 6.0)and Chinese Dependency Treebank as the source Treebank [5].

Michael proposed three statistical parsing model, which is a generative model of lexicalized context-free grammar. They used Wall Street Journal as training Treebank [6].

Our work focuses on the unification of CFG rules for multiple treebanks in order to unification of these treebanks in one Treebank. We use, also, large annotated corpus as lexical resource for lexical rules in the Treebank because must treebankssuffer from sparse data.

Statistical parsing and probabilistic CFG (PCFG)

Almost any natural language sentence is ambiguous in structure as shown in figure 1 where the sentence has two syntactic meaning. The left parse can interpret the sentence as “the dog is holding the telescope” but the right parse tree interpret the sentence as “the man is using the telescope to see the dog” [7]. The parsing process depends on the application that need to it. For example in Grammar checking application need to check the sentence has parse tree or not. In such case, one parse tree (any one) is sufficient. In machine translation, we need to all parse tree and selecting the best one. Selecting the best tree cannot be estimated from the traditional CFG but can be calculated using PCFG. A Treebank, collection of text with their parse tree, is used for calculating PCFG.

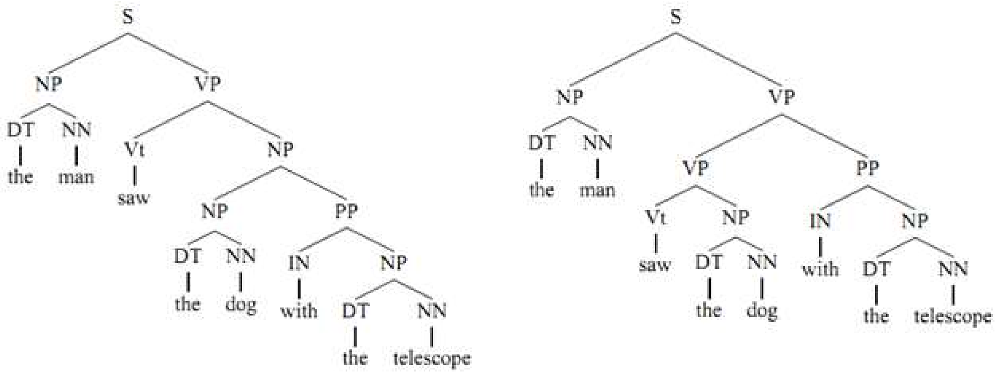


Figure (1)Two parse trees (derivations) for the sentence the man saw the dog with the telescope [7].

Probabilistic context-free grammar (PCFG) is defined to be the form G = (N, Σ, S, R, p) where [8]:

1- G = (N, Σ, S, R) is a context-free grammar
2- p is the probabilities parameter:
p(A→ β): where A is one non-terminal

for each rule A→ β ∈ R. The parameter p(A→ β) can be interpreted as the conditional probabilty of choosing rule A→ β in a left-most derivation, given that the non-terminal being expanded is A. For any X ∈N , we have the constraint:

$$\sum_{A \to \beta \in R : A = X} p(A \to \beta) = 1$$

where p(A → β) ≥ 0 for any A → β ∈ R. This simply states that for any non-terminal X , the parameter values for all rules with that non-terminal on the left-hand-side of the rule must sum to one.for example if we have the following rules for VP:
VP→V (0.3)
VP→V NP (0.7)
Then the sum of the probabilities equal to one:P(VP→V) + P(VP→V NP)=1
The probabilitiesare estimated from the Treebank using:

$$p(A \to \beta) = \frac{Count(A \to \beta)}{Count(A)}$$

Where Count (A → β) is the number of times that the rule A → β is seen in the treebank, and Count(A) is the number of times the non-terminal A is seen in the Treebank. For example, if the rule VP → V NP is seen 70 times in our corpus, and the non-terminal VP is seen 100 times, then $P(VP \to \mathrm{V\ NP}) = \frac{70}{100} = 0.7$.
We can estimate the probability of the whole sentence tree (t) by multiplying the probabilities of its components rules. Suppose that we have parse tree t ∈ T_G containing rules $A_1 \to \beta_1$, $A_2 \to \beta_2$, . . . , $A_n \to \beta_n$, the probability of t under the PCFG is:

$$p(t) = \prod_{i=1}^{n} p(A_i \rightarrow \beta_i)$$

For the trees in figure 1 the final probabilities can be as following:

$$\begin{aligned} p(t_{left}) = {} & P(S \rightarrow \text{Np Vp}) \times P(NP \rightarrow \text{DT NN}) \times \text{P(DT} \rightarrow \text{the)} \times \text{P(NN} \rightarrow \text{man)} \times \text{P(VP} \\ & \rightarrow \text{Vt NP)} \times \text{P(Vt} \rightarrow \text{saw)} \times \text{P(NP} \rightarrow \text{NP PP)} \times P(NP \\ & \rightarrow \text{DT NN}) \times P(PP \rightarrow \text{IN NP}) \times P(IN \rightarrow \text{with}) \times P(NP \rightarrow \text{DT NN}) \times \text{P(DT} \\ & \rightarrow \text{the)} \times \text{P(NN} \rightarrow \text{telescope)} \\ p(t_{right}) = {} & P(S \rightarrow \text{Np Vp}) \times P(NP \rightarrow \text{DT NN}) \times \text{P(DT} \rightarrow \text{the)} \times \text{P(NN} \rightarrow \text{man)} \times \text{P(VP} \\ & \rightarrow \text{VP PP)} \times \text{P(VP} \rightarrow \text{Vt NP)} \times \text{P(Vt} \rightarrow \text{saw)} \times P(NP \\ & \rightarrow \text{DT NN}) \times P(PP \rightarrow \text{IN NP}) \times P(IN \rightarrow \text{with}) \times P(NP \rightarrow \text{DT NN}) \times \text{P(DT} \\ & \rightarrow \text{the)} \times \text{P(NN} \rightarrow \text{telescope)} \end{aligned}$$

The tree which gives the maximum probability will be selected.

The used Treebanks

Three Treebanks are used in our work: **British National Corpus** (BNC 1000), GENIA Treebank (GENIA_treebank_v1) and part of Penn Treebank PTB (199 syntactic parsed articles from PTB) freely available.

British National Corpus (BNC 1000):

The BNC consist of one hundred million words of British English. It is balanced corpus. 90% of it is written text and 10% consists of transcribed spontaneous and scripted spoken language [9].
BNC 1000 is Gold Standard Parse Trees for 1,000 sentences from the British National Corpus annotated according to Penn Treebank bracketing guidelines and checked using Markus Dickinson's Treebank annotation error detection software [10].
We see in BNC Treebank there is a production like this ". → ." we changed it to "DOT → ." and any occurrence of the dot "." In the productions were replaced by "DOT". For example The production "S → NP VP ." are replaced to "S → NP VP DOT". The same work was done with the same situations for other special symbols.

GENIA Treebank (GTB)

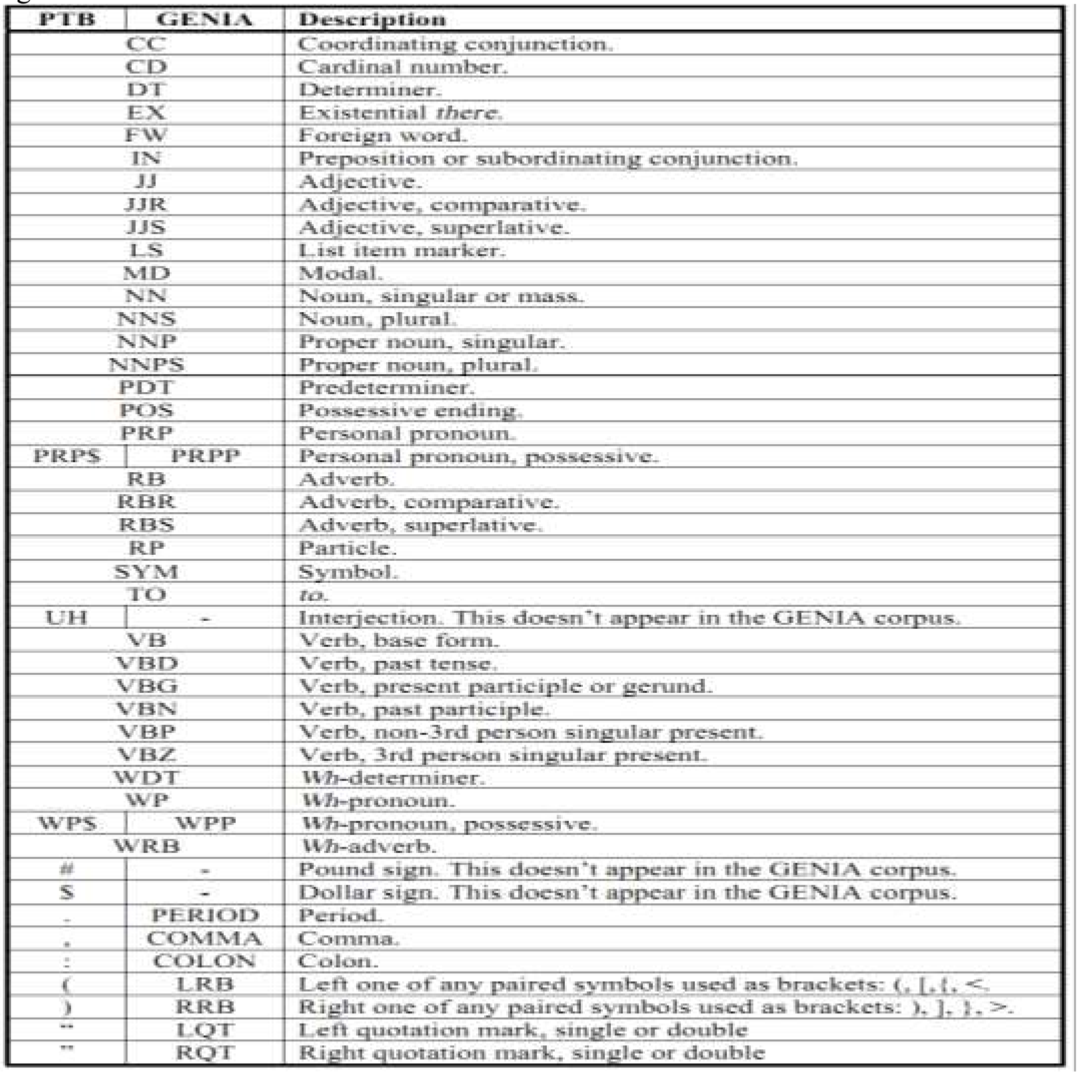
The primary GENIA corpus is made up of the titles and abstracts of journal articles which have been taken from the Medline database [11].
It consists of 1999 abstracts annotated with Part-of-speech and syntactic (phrase structure) annotation. It is distributed in XML format. Some researchers convert it to PTB format but we did not find this format; therefore we converted it into PTB format by writing simple program.
The used tagset of GENIA corpus consist of 45 tags. These tags used as pre-terminals. There are 23 Non-terminals in GENIA Treebank.

Penn Treebank (PTB)

English Penn Treebank is Standard corpus for testing syntactic parsing consists of 1.2 M words of text from the Wall Street Journal (WSJ). PTB containsabout 42,416 articles. There are 199 parsed articles freely available(WSJ0001- WSJ0199) in NLTK package .The Penn Treebank II (PTB) bracketing guidelines are shown in [12]. The used tagset is 36 tags and12 symbols of punctuations and special symbols.

The proposed Unification of POS tags and syntactic tags for the three Treebanks

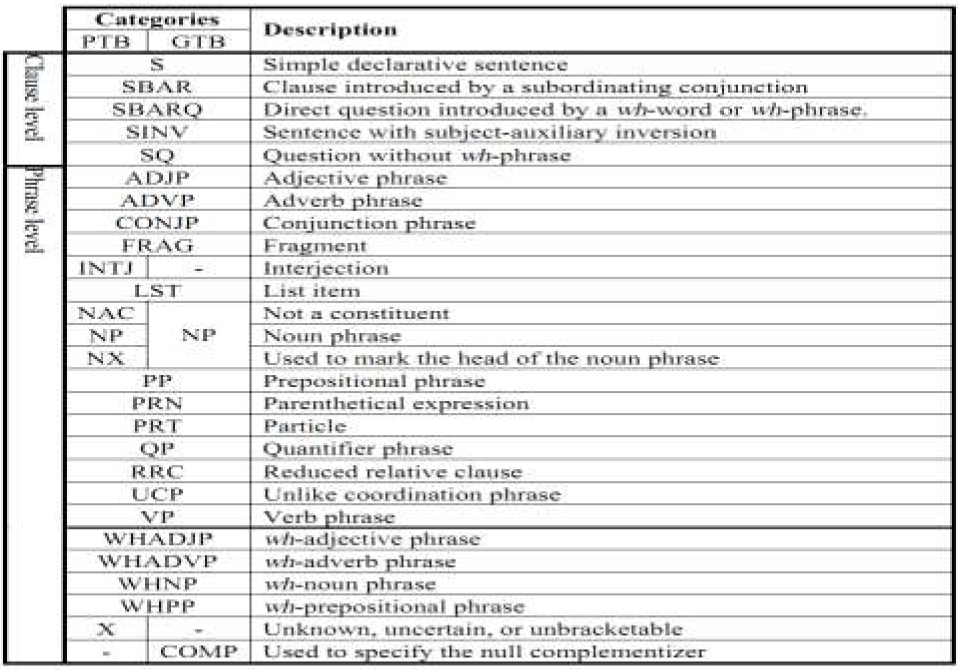
We, firstly, unified the tagsets of the three treebanks. This operation is called mapping where it was done manually. We select the small size tagset and mapping the others to it. The useful thing is that the tagsets of these Treebanks are much closed which simplify the mapping work. BNC-1000 treebank used PTB tagset and style, therefore, it can be as PTB. Therefore we unified these tagsets to GENIA tagset without any sophisticated mapping process, see figure 2 [11].There are simple modifications where adding little tags as HASH tag for "#" and SGND tag for "$" and so on.



<table>
<tr><th>PTB</th><th>GENIA</th><th>Description</th></tr>
<tr><td colspan="2">CC</td><td>Coordinating conjunction.</td></tr>
<tr><td colspan="2">CD</td><td>Cardinal number.</td></tr>
<tr><td colspan="2">DT</td><td>Determiner.</td></tr>
<tr><td colspan="2">EX</td><td>Existential <i>there</i>.</td></tr>
<tr><td colspan="2">FW</td><td>Foreign word.</td></tr>
<tr><td colspan="2">IN</td><td>Preposition or subordinating conjunction.</td></tr>
<tr><td colspan="2">JJ</td><td>Adjective.</td></tr>
<tr><td colspan="2">JJR</td><td>Adjective, comparative.</td></tr>
<tr><td colspan="2">JJS</td><td>Adjective, superlative.</td></tr>
<tr><td colspan="2">LS</td><td>List item marker.</td></tr>
<tr><td colspan="2">MD</td><td>Modal.</td></tr>
<tr><td colspan="2">NN</td><td>Noun, singular or mass.</td></tr>
<tr><td colspan="2">NNS</td><td>Noun, plural.</td></tr>
<tr><td colspan="2">NNP</td><td>Proper noun, singular.</td></tr>
<tr><td colspan="2">NNPS</td><td>Proper noun, plural.</td></tr>
<tr><td colspan="2">PDT</td><td>Predeterminer.</td></tr>
<tr><td colspan="2">POS</td><td>Possessive ending.</td></tr>
<tr><td colspan="2">PRP</td><td>Personal pronoun.</td></tr>
<tr><td>PRP$</td><td>PRPP</td><td>Personal pronoun, possessive.</td></tr>
<tr><td colspan="2">RB</td><td>Adverb.</td></tr>
<tr><td colspan="2">RBR</td><td>Adverb, comparative.</td></tr>
<tr><td colspan="2">RBS</td><td>Adverb, superlative.</td></tr>
<tr><td colspan="2">RP</td><td>Particle.</td></tr>
<tr><td colspan="2">SYM</td><td>Symbol.</td></tr>
<tr><td colspan="2">TO</td><td><i>to</i>.</td></tr>
<tr><td>UH</td><td>-</td><td>Interjection. This doesn't appear in the GENIA corpus.</td></tr>
<tr><td colspan="2">VB</td><td>Verb, base form.</td></tr>
<tr><td colspan="2">VBD</td><td>Verb, past tense.</td></tr>
<tr><td colspan="2">VBG</td><td>Verb, present participle or gerund.</td></tr>
<tr><td colspan="2">VBN</td><td>Verb, past participle.</td></tr>
<tr><td colspan="2">VBP</td><td>Verb, non-3rd person singular present.</td></tr>
<tr><td colspan="2">VBZ</td><td>Verb, 3rd person singular present.</td></tr>
<tr><td colspan="2">WDT</td><td><i>Wh</i>-determiner.</td></tr>
<tr><td colspan="2">WP</td><td><i>Wh</i>-pronoun.</td></tr>
<tr><td>WP$</td><td>WPP</td><td><i>Wh</i>-pronoun, possessive.</td></tr>
<tr><td colspan="2">WRB</td><td><i>Wh</i>-adverb.</td></tr>
<tr><td>#</td><td>-</td><td>Pound sign. This doesn't appear in the GENIA corpus.</td></tr>
<tr><td>$</td><td>-</td><td>Dollar sign. This doesn't appear in the GENIA corpus.</td></tr>
<tr><td>.</td><td>PERIOD</td><td>Period.</td></tr>
<tr><td>,</td><td>COMMA</td><td>Comma.</td></tr>
<tr><td>:</td><td>COLON</td><td>Colon.</td></tr>
<tr><td>(</td><td>LRB</td><td>Left one of any paired symbols used as brackets: (, [, {, <.</td></tr>
<tr><td>)</td><td>RRB</td><td>Right one of any paired symbols used as brackets:),], }, >.</td></tr>
<tr><td>``</td><td>LQT</td><td>Left quotation mark, single or double</td></tr>
<tr><td>''</td><td>RQT</td><td>Right quotation mark, single or double</td></tr>
</table>

Figure (2): POS tagset of PTB and GTB which can be the unified tagset.

The three Treebanks are used same syntactic symbols with very little difference. We can see figure 3 where the unification is straight forward.



	Categories		Description
	PTB	GTB	
Clause level	S		Simple declarative sentence
	SBAR		Clause introduced by a subordinating conjunction
	SBARQ		Direct question introduced by a *wh*-word or *wh*-phrase.
	SINV		Sentence with subject-auxiliary inversion
	SQ		Question without *wh*-phrase
Phrase level	ADJP		Adjective phrase
	ADVP		Adverb phrase
	CONJP		Conjunction phrase
	FRAG		Fragment
	INTJ	-	Interjection
	LST		List item
	NAC	NP	Not a constituent
	NP		Noun phrase
	NX		Used to mark the head of the noun phrase
	PP		Prepositional phrase
	PRN		Parenthetical expression
	PRT		Particle
	QP		Quantifier phrase
	RRC		Reduced relative clause
	UCP		Unlike coordination phrase
	VP		Verb phrase
	WHADJP		*wh*-adjective phrase
	WHADVP		*wh*-adverb phrase
	WHNP		*wh*-noun phrase
	WHPP		*wh*-prepositional phrase
	X	-	Unknown, uncertain, or unbracketable
	-	COMP	Used to specify the null complementizer

Figure (3): Syntactic Nonterminals of the grammars used by GTB and PTB

Improving lexical probabilities

Parsing has problem of sparse data especially for very small lexical which constructed from the Treebank. Using any word outside of Treebank will cause errors in parsing which known as unknown words problems. For improving performance of our work, we add **lexical rules**[1] constructed from Brown corpus. Almost all researchers made the lexical entries from the tested Treebank. We achieved that by adding lexical productions in the Treebank for all the words in the lexical. For example if we found "book NN" in the corpus, we add a rule "NN→book".The lexical classes are unified manually (mapping) with the all three Treebanks. It was hard work but it is done. We apply the following rules for mapping between Brown Tagset and Unified tagset, the mapping is shown in table-2:

1. *IF Tag Tn in Brown Tagset has one to one mapping tag, give it the match tag*
2. *IF Tag Tn in Brown Tagset has one to multiple tags mapping, give it all the match tag*
3. *IF Tag Tn in Brown Tagset has multiple to one tag mapping, give all of this tag the match tags.*

See Figure (4. The real problem occurs only with t_{2b} because it has two match classes.

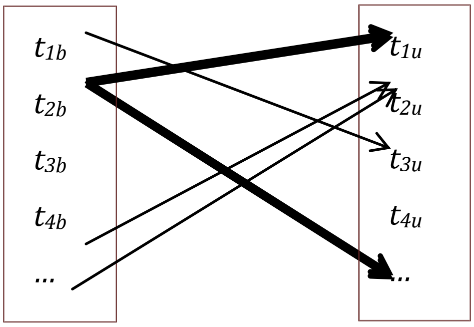


Figure (4):mapping of two tagsets

[1] Lexical rules are special rules in CFG which has the form Nonterminal→ terminal

We manipulate it as in rule 2 where t_{2b} is replaced by t_{1u} and t_{nu}. This will cause problem in counting probabilities of the words classes for lexical. It can be solved by dividing, for example, number of occurrence of t_{2b} by number of matched classes (2 in our example fort_{1u} and t_{nu}).
After doing mapping between Brown tagset and the unified Tagset, the lexical probabilities are calculated from Brown corpus and the three Treebanks. This is done by neglecting all rules in the Treebanks except the lexical rules (a rule have a non-terminal on left side and one terminal in right side). We should see that these probabilities do not affect the whole probabilities of CFG because we changed only lexical probabilities which will help the parser for recognize large scale of words (decreasing of possibility of appearing of unknown words). We can see that any Treebank (or CFG rules) has two types of rules[2]:
$A \rightarrow$ αwhereα any combination of Non-terminals and terminals. Or
$A \rightarrow$ twhere t is any terminal this rule is called lexical rule because it has the POS for each word

Table (1) Brown tagset (simplified) and the corresponding tags in the unified tagset

Brown Tag	Brown tag meaning	Corresponding Tags in the unified tagset
ADJ	Adjective	JJ
ADV	Adverb	RB, RBR, RB$,
CNJ	Conjunction	CC
DET	Determiner	DT, PDT
EX	Existential	EX
FW	foreign word	FW
MOD	modal verb	MD
N	Noun	NN
NP	proper noun	NNP,NNPS
NUM	Number	CD
PRO	Pronoun	PRP, PRP$, PRPP,
P	Preposition	IN
TO	the word to	TO
UH	Interjection	UH
V	Verb	VB,VBP, VBZ
VD	past tense	VBD
VG	present participle	VBG
VN	past participle	VBN
WH	wh determiner	WDT, WP,WP$, WPP,WRB
.	dot	PERIOD
,	comma	COMMA
(	opening parenthesis	LRB
)	closing parenthesis	RRB
:	:	COLON
"	"	DQ (LQT,RQT)

Unknown words Processing:
When we train the treebanks there are little unknown words because we used large lexical resource built from Brown corpus. These unknown words cause problems in parsing. We solved this problem by using two steps: firstly we add special rules to the main four POSs as Noun, Verb, Adverb and Adjective. Secondly, we use simple analyzer to choose one of these main four POSs.
Step1: We add four rules to these POSs in the Treebank with very small probabilities likes :
N→xxxxxx (0.00000000001)
Adv → yyyyyy (0.00000000001)
Adj → zzzzzz (0.00000000001)

[2] epsilon production (ε) is not used in Treebank.

V → mmmmmm (0.00000000001)

Step2: The analyzer will choose the main POSaccording to the following rules:

1- By default the word is Nounif none of the following rules not applicable.
2- If the word ends with {ed, ing, en, ize, …} then it is verb.
3- If the word ends with {al, able, less, ly, …} then it is adverb.
4- If the word ends with {er, est, ive, ative, ful, , …} then it is adjective.

If the unknown word is Noun then we replace “xxxxxx” in the rule “N→xxxxxx” by this word and no anything else. The Treebank is contains this word now and the parser will work normally.

Implementation and results

We, firstly, took each treebank and we did train-test for it separately. This is done after unification of tagsets and syntactic symbols (Non-terminals in CFG). The used parser is Viterbi parser which is tool in NLTK package in Python environment. The results of BNC, GTB and PTB are shown in Table 2. The same test is done using the modified Treebanks where the results shown in Table 3.

The third test is done using all the Treebanks collectively after their unification without using Brown corpus as lexical resource.The results of this test are shown in Table 4.The fourth test is done same as the third test with using Brwon corpus as lexical resource as shown in Table 5.

We used Precision, Recall and F-measure as measuring factors. Where, Precision is the number of correct constituents produced by the parser divided by the total number of constituents produced by the parser. Recall is the number of correct constituents produced by the parser divided by the total number of constituents in the set of gold standard parse trees. The f-measure is the harmonic mean of precision and recall [9]:

$$P = \frac{\#\text{correct constituents}}{\#\text{constituents produced by parser}}$$

$$R = \frac{\#\text{correct constituents}}{\#\text{constituents in gold standard}}$$

$$F = 2\frac{Precision\,.Recall}{Precision + \ Recall}$$

Table(2) training the three treebanks separately before modification

Treebank	Overall sentences	#constituents in manually parsing	#constituents in System parsing	#correct constituents in System	R	P	F1
BNC	1000	51858	51600	42687	82.31517	82.72674	82.52044
PTB	3914	179413	178286	150448	83.85568	84.38576	84.11989
GTB	20540	899390	894754	763857	84.93056	85.37062	85.15002
Over all (average)							**83.93011**

Table (3) training the three treebanks separately after modification

Treebank	Overall sentences	#constituents in manually parsing	#constituents in System parsing	#correct constituents in System	R	P	F1
BNC	1000	51858	51503	42233	81.4397	82.00105	81.71941
PTB	3914	179413	178092	147578	82.25602	82.86616	82.55996
GTB	20540	899390	891952	755050	83.95134	84.65142	84.29993
Over all (average)							**82.85976**

Table (4) training the three treebanks collectively after modification without support of lexical resource

Treebank	**Overall sentences**	#constituents **in manually parsing**	#constituents **in System parsing**	**#correct** constituents **in System**	**R**	**P**	**F1**
All Treebanks Collectively	25454	1130661	1122212	961188	85.01116	85.6512	**85.32998**

Table (5) training the three treebanks collectively after modification with support of lexical resource

Treebank	**#sentences**	#constituents **in manually parsing**	#constituents **in System parsing**	**#correct** constituents **in System**	**R**	**P**	**F1**
All Treebanks Collectively	25454	1130661	1123587	996803	88.16108	88.71614	**88.43774**

Discussion and Future work

The size of the trained data, treebank in our work, is very important in performance of parser in statistical parsing. There are many treebanks, for English language, available for free but each one has private CFG rules and POS tagset. Also, the size of each Treebank is very small comparing with the annotated corpora because it needs more manually work from experts in the field of linguistics.

As we see, our work start by unification of POS tagsets of the three Treebank then the mapping process between Brown Corpus tagset and the unified tagset is done. This is done manually according to our experience in this field. Also, all the non-terminals in the CFG production are unified. All the three Treebanks and the Brown corpus are rebuilt according to the new modification.

Four types of tests are made for the proposed unification. As we can see, in our paper, the unification of multiple treebanks can increase the accuracy resulting from increasing size of the trained data. But if we test each Treebank separately after unifying Non-terminals and terminals in CFG, the correct constituents is little bit less than before modification in some test are seen Table(2&

Table (3. It can be interpreted by **cumulative errors**[3] as a result of unification of the symbols. But if these treebanksare collected in one Treebank then this errorswill be eliminated because of numbers of examples and then effective probabilities values, as shown in Table (4.

Adding annotated corpus as lexical resource for the Treebank decreased the possibility of occurring unknown words. Also, the probabilities of lexical rules are near from reality lead to addition more accuracy as inTable 5.

[3] We mean by cumulative errors that if the errors occur in one tree result from a wrong nonterminal production, then some of the above trees will be wrong.

Using very simple approach, for selecting the POS of unknown words (lexical rule),decreases the errors. We used this simple approach in all tests of Table(2 to Table 5. We used four POSs (N, ADV, ADJ and V) for unknown words which are the main POS instead of selecting A Noun POS for each unknown word. This is, surely, improve parser performance.

We can show, as summary, that (a) the unification of multiple Treebanks can be done and will increase the accuracy. (b) A large annotated corpus as Brown corpus can be used for (i) decreasing the unknown words and (ii) extracting the probabilities nearest to the reality. (c) The mapping between the unified tagset and the lexical tagset (used in Brown corpus) can be done straightforward.

The novelty of our work can be addressed in three factors: (i) unification of these three treebanks, (ii) using annotated corpus as lexical resource from outside of Treebank, and (iii) method of processing of unknown words for four parts of POS tagset.

REFERENCES

[1]Marie-Catherine De Marneffe, Bill MacCartney, Christopher D Manning, and others, "Generating typed dependency parses from phrase structure parses," in Proceedings of LREC, vol. 6, 2006, pp. 449-454.

[2] Sampo Pyysalo et al., "On the unification of syntactic annotations under the Stanford dependency scheme: A case study on BioInfer and GENIA," in Proceedings of the Workshop on BioNLP 2007: Biological, translational, and clinical language processing, 2007, pp. 25-32.

[3] Muhua Zhu and Jingbo Zhu, "Automatic treebank conversion via informed decoding," in Proceedings of the 23rd International Conference on Computational Linguistics: Posters, 2010, pp. 1541-1549.

[4] Zheng-Yu Niu, Haifeng Wang, and Hua Wu, "Exploiting heterogeneous treebanks for parsing," in Proceedings of the Joint Conference of the 47th Annual Meeting of the ACL and the 4th International Joint Conference on Natural Language Processing of the AFNLP: Volume 1-Volume 1, 2009, pp. 46-54.

[5] Zhenghua Li, Ting Liu, and Wanxiang Che, "Exploiting multiple treebanks for parsing with quasi-synchronous grammars," in Proceedings of the 50th Annual Meeting of the Association for Computational Linguistics: Long Papers-Volume 1, 2012, pp. 675-684

[6] Michael Collins, "Three generative, lexicalised models for statistical parsing," in Proceedings of the 35th Annual Meeting of the Association for Computational Linguistics and Eighth Conference of the European Chapter of the Association for Computational Linguistics, 1997, pp. 16-23

[7] Michael Collins, "Probabilistic Context-Free Grammars (PCFGs)," Lecture Notes, 2013.

[8] Alexander Clark, Chris Fox, and Shalom Lappin, The handbook of computational linguistics and natural language processing.: John Wiley \& Sons, 2013.

[9] Jennifer Foster and Josef Van Genabith, "Parser evaluation and the bnc: Evaluating 4 constituency parsers with 3 metrics," 2008.

[10] Markus Dickinson, "Ad Hoc Treebank Structures.," in ACL, 2008, pp. 362-370.

[11] Jin-Dong Kim, Tomoko Ohta, Yuka Teteisi, and Jun'ichi Tsujii, "GENIA Corpus Manual," 2006.

[12] Ann Bies et al., "Bracketing guidelines for Treebank II style Penn Treebank project," University of Pennsylvania, vol. 97, p. 100, 1995.