Natural Language Analysis – COMP 6751

Project 2

Name: Sujith Manikandan

Student ID: 40302479

Abstract and Motive

The objective of this project is to turn the context free grammar developed in project one into a feature grammar. Feature grammar makes uses of features along with the terminals and non-terminals for the parsing of the sentences. The main problem with the CFG is that the grammar only checks for the syntax of the sentence and not for the semantics. That is, it cannot differentiate between grammatically correct and ungrammatical sentences. The use of feature grammar restricts the parsing of ungrammatical sentences by having proper rules defined with the features.

Preprocessing Pipeline

The preprocessing pipeline to extract the sentences from the sgml file with annotations and then split the data into sentences and words is the same as the one used in project 1.

Design For Grammar

There are various features that define the semantics of a language and some of the top most important features are the number (singular/ plural), PERSON (1,2,3) and Tense (Past, Future, Present)

The feature grammar developed as a part of this project ensures the proper use of number agreement for the nouns (Noun Phrases) and the number and person agreement for the verbs and pronouns (Verb Phrases) present in sentences.

Thus, this grammar unlike the previous CFG doesn't allow ungrammatical sentences up to a certain extent.

Part1 – Subcategorization of Terminals

Part a) Noun Phrases – Nouns and Determiners

In order for the words to be matched with the features, the terminals list present in the CFG is split into sub lists based on features. The list of all the nouns is separated as singular nouns and plural nouns. Eg:

N[NUM=sg] -> "tablet", "pen", "paper", "foot"

N[NUM=pl] -> “medicines”, “pens”, “feet”

All the nouns without a number is taken into the common noun class. Eg,

N[NUM=?n] -> “happiness”, “hunger”

The ‘?n’ operator is used to match any feature without specification.

Then there are the determiners, it is the combination of a proper determiner and the noun that determine a valid noun phrase. All singular determiners are to occur with singular nouns and plural determiners with the plural nouns.

Eg, ‘this dog’ and ‘these dogs’ are the grammatically valid phrases in English and not

‘these dog’ and ‘this dogs’. Now that the nouns are classified, the determiners must also be classified next.

det[NUM=sg] -> "this" | "that" | "each" | "every" | "another" | "either" | "an" | "a" | "any" | "some" | "no" | "the"

det[NUM=pl] -> "these" | "those" | "both" | "all" | "any" | "some" | "no" | "half" | "the"

In the above set, there are a few determiners such as ‘this’, ‘that’, ‘each’ and ‘every’ that occur only with singular nouns and a few other of them such as ‘these’, ‘those’ and ‘both’. Certain determiners such as ‘the’, ‘some’ and ‘any’ can be both singular and plural. Eg, ‘the book’ and ‘the books’, ‘any idea’ and ‘any ideas’.

Part b) Verb Phrases – Verbs and Pronouns

Similarly, in order to ensure the number and person agreement with the verbs, the verbs and the pronouns are separated based on both the features. The number feature has two values (singular and plural) and the person feature has 3 values (1,2,3) thus together there are six different combinations possible for these two features.

The pronouns are subcategorized as follows,

pronoun [PERSON=1, NUM=sg] -> "I" | "my"

pronoun [PERSON=1, NUM=pl] -> "we" | "our" | "us"

pronoun [PERSON=2, NUM=? n] -> "you" | "your" | "yourself"

pronoun [PERSON=3, NUM=sg] -> "he" | "his" | "she" | "her" | "it" | "its"

pronoun [PERSON=3, NUM=pl] -> "they" | "them" | "their"

When it comes to the verbs, certain combinations tend to work in a similar way. For eg, the group of all the plural verbs irrespective of the person are the same.

V [NUM=sg, PERSON=1] -> "am" | "was" | "have" | "become" | "let" | "advice" | "want"

V [NUM=sg, PERSON=2] -> "were" | "are" | "have" | "become" | "let" | "advice" | "want"

V [NUM=pl, PERSON=?n] -> "were" | "are" | "have" | "become" | "let" | "advice" | "want"

V [NUM=sg, PERSON=3] -> "is" | "was" | "has" | "becomes" | "lets" | "advices" | "wants"

The auxiliary verb ‘am’ is only used with singular first person ‘I’ and it doesn’t occur with any other pronouns. Similarly, is occurs with singular third person pronouns ‘he’ ‘she’, ‘it’ and other proper nouns. The was is the past form of the verbs ‘am’ and ‘is’ and thus is used with the same pronouns that use these verbs.

The words ‘were’ and ‘are’ are the past and present form for the plural nouns. Thus, irrespective of the PERSON attribute, these words are used for all the plural words, and it is denoted with the use of the ?n operator. **One special case in English is that the second person singular verbs behave exactly as similar to the plural verb forms mentioned above.** This occurs because the words ‘you’ and ‘your’ can be interpreted as both singular and plural in English. This is the reason why the singular context of the words ‘you’ and ‘your’ also takes the context of the plural ones.

Then the word have is used for the plural form of verbs and also by the singular second person. **However, the special case with have is that it also applies to first person singular in addition to the above-mentioned values. Thus, the third person singular is the only category that uses ‘has’ instead of ‘have’.**

Although the separation among the categories were complex for the auxiliary verbs, the separation for all the other verbs is very simple and it is just a binary separation. All 5 categories except for the one deal with the verbs in their base form. Only the category of third person singular deals with a special form which adds the letter ‘s’ or ‘es’ to the base form of the verb.

Eg, I do | We do | You do | They do --- He do**es** | She do**es** | It do**es** | Sujith do**es** (proper noun)

I want | We want | You want | They want --- He want**s** | She want**s** | Sujith want**s** (proper noun)

This case is true for all the non-auxiliary verbs in English. Apart from these, there are certain forms of the verbs, the gerund and the past tense forms that doesn’t require any subcategorization. All the categories take the same form of the gerund and the past tense verbs.

Eg You are running | he is running | they are running |

You ran | I ran | they ran | she ran

For the task of named entity detection all the words belonging to the following three categories are grouped together in the name of its terminal → Drugs, Chemicals and Organization.

Part 2 – Non-Terminals

Now that the list of terminals have been successfully categorized, the non-terminal needs to be changed in such a way that the valid categories occur together and no invalid categorical combination occurs.

The changes has to happen to the noun phrases, verb phrases, prepositional phrases and the final sentence. For the prepositional phrases and sentences, no need to set any specific categories, the value of the categorized terminals can be general [NUM=?n, PERSON=?p].

For the noun phrases, it is necessary to ensure that the number of the determiner and the nouns are equal.

Det[NUM=?n] Noun[NUM=?n] is passed and the Earley feature grammar would understand that whatever is the number of the determiner, the same must also match for the noun, otherwise the combination is neglected and is not parsed.

Also, for the named entity detection, the terminals of the named entities are also present in the place of a noun in the noun phrases.

NP → Chemical | Drug | Chemical NP | Drug NP | Org Org | Org NP

Any organization is a minimum of two words with a word such as ‘ltd’ ‘trading’ ‘private’ ‘pharmaceuticals’ and ‘store’ occurring together.

The verb phrases are also similarly grouped based on the person and number of the pronouns, nouns and the verbs.

VP2[NUM=?n,PERSON=?p] -> V[NUM=?n,PERSON=?p] | V[NUM=?n,PERSON=?p] NP[NUM=?n] | pronoun[NUM=?n,PERSON=?p] V[NUM=?n,PERSON=?p]

Conjugations And Concatenations

The conjugations and concatenations are the special cases were the forms of the phrases can intermix together. For eg, Noun[Plural] conj Noun[Singular]

VP[plural] conj VP[Singular]

‘the dogs are barking and he is watching them’

Thus, while writing the rules for conjugation, it is necessary to ensure that all the combination of attributes occur in both the sides of the conjugation.

Demerits of the Grammar

Although the feature grammar is better when compared to the Context Free Grammar, it is still a long way from being perfect. This is because the developed grammar only contains two attributes, number and person and there are so many other important attributes like tenses and gender.

Eg 'I give her the book yesterday' In this sentence, 'I give' is correct as give is the verb in the same category of 'I' singular first person. However, giving her the book yesterday is not possible as the word give is 'present' tense and yesterday refers to 'past' tense. Thus, the word give is to be replaced with its past form 'gave'. However, without the tense attribute, the checking for this mistake is not possible.

Similarly, 'He is the topper of the class, and this is her project' in this sentence the first half of the sentence tells that the topper of the class is a male (he) and in the second half (her) the female pronoun is used to refer to the same person and that is not possible.

Another example 'books are playing' in this case the noun books and the auxiliary verb 'are' are plural, it doesn't make sense as book doesn't have a gender and it cannot 'play' as the material has no life. Thus, these nouns can only be used as an object and not as a subject of the sentence.

Then we have the transitive and intransitive verbs category which can have two nouns (subject and object) or just one noun (subject). Eg 'He sneezed the book'.

In addition to these there are so many different features that need to be added to the grammar set and there are so many other combinations as well that can be present between the categories of the grammar. Unfortunately, due to computational constraints, not all combinations of the categories could be used in the non-terminal part of the grammar.

Resolving of ambiguity is another important aspect of a good grammar that is not present in the developed grammar. The grammar also doesn't take the dependency relations into account.

Eg "she sings better than he" In this sentence, the features are properly in place and yet the pronoun he has to be replaced with the pronoun him. As we do not have any categorization for the pronouns being a subject and a object.

Results and Future Scope

The grammar developed as a part of this project is able to parse about 65% of the sentences extracted from the six main PIL leaflets. It also provides nearly similar results with other leaflets in the PIL corpus as well.

As mentioned, the addition of more features such as gender, tense and transitivity can increase the validation of the sentences to a much greater extent reducing the scope of parsing ungrammatical sentences.

Another important aspect is increasing the values of the terminals. This grammar only contains the terminals extracted from the PIL corpus and no other extra words are present apart from a few pronouns and words added manually. Thus, we could increase the number of words. However, it takes more effort and time to split the words into their subcategories as it needs to be done manually. With more features added, the subcategories increases and the complexity of the non-terminal part of the grammar highly increases.

The feature grammar can also be extended to more languages apart from English. We would have to include the terminal words from other languages and use separate non-terminal phrases for those languages.

The named entity recognition can be extended to all other entities such as name of a person, time, date, and location. Then it is also possible to do thematic/ semantic role assignment to words in the sentences such as the main theme, goal and agent.

Then the feature grammar could also solve the problem of context sensitive semantics. For eg, the references of pronouns in the sentences can be mapped to the right subject with proper features in use.

Then it is also possible to make the feature grammar understand logical statements. Eg ‘If it rains, I take the umbrella’. As given in this example the if, iff and all other logical statements’ semantics could be resolved.

Thus, feature grammars in general allow the moving from a context free to a context sensitive language to make sense of the semantics and validate their occurrence along with the syntax.