Chapter 6

Generation Component

In this chapter we describe the generation component which succeeds the translation component. Generation Component constructs the target language translation of the original source sentence from the intermediate forms and then applies some grammatical corrections on the output to increase its accuracy. It consists of two modules, Synthesis and Post Processing. A detailed description of both the modules is provided in this chapter.

6.1 Synthesis

Synthesis phase carries out the reverse of analysis phase and transforms the strings of English words into grammatically correct sentences. The output of the transfer component is given as input to synthesis phase. The stage of synthesis adapts the extracted Target Language fragments and combines them into Target Language sentences. In general, the synthesis of target language (TL) sentences is less complex than the analysis of source language (SL) input. Synthesis must convert the phrase structure into an appropriate TL structure and produce a linear representation i.e. the inversion of the analysis process in some way. However, it should be stressed that inversion does not imply that the rules devised for the analysis of structures for a particular language (as SL) can be simply reversed to obtain rules for synthesis of that language (as TL).
The Synthesis component applies a number of simple linguistic transformations so that the translated English phrases form an English sentence. Transformation rules are essential when the structure of source and target languages are different. Table 6.1 shows the design of the database for storage of rules for synthesizing target language sentences.

**Table 6.1 Database Design for Synthesis Phase**

<table>
<thead>
<tr>
<th>Field Name</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>0</td>
<td>Start phrase</td>
</tr>
<tr>
<td>1</td>
<td>First phrase</td>
</tr>
<tr>
<td>2</td>
<td>Second Phrase</td>
</tr>
<tr>
<td>3</td>
<td>Third Phrase</td>
</tr>
<tr>
<td>4</td>
<td>Fourth phrase</td>
</tr>
<tr>
<td>5</td>
<td>Fifth Phrase</td>
</tr>
<tr>
<td>Sen0</td>
<td>In target language, Serial number of phrase at start</td>
</tr>
<tr>
<td>Sen1</td>
<td>Serial number of the phrase positioned first</td>
</tr>
<tr>
<td>Sen2</td>
<td>Serial number of phrase at second position</td>
</tr>
<tr>
<td>Sen3</td>
<td>Serial number of phrase at third position</td>
</tr>
<tr>
<td>Sen4</td>
<td>Serial number of phrase at fourth position</td>
</tr>
<tr>
<td>Sen5</td>
<td>Serial number of phrase at fifth position</td>
</tr>
</tbody>
</table>
Some Sample Entries for Synthesis of Sentences

<table>
<thead>
<tr>
<th></th>
<th>1</th>
<th>2</th>
<th>3</th>
<th>4</th>
<th>5</th>
<th>sen0</th>
<th>sen1</th>
<th>sen2</th>
<th>sen3</th>
<th>sen4</th>
<th>sen5</th>
</tr>
</thead>
<tbody>
<tr>
<td>np</td>
<td>np</td>
<td>pp</td>
<td>vp</td>
<td>-</td>
<td>-</td>
<td>0</td>
<td>3</td>
<td>1</td>
<td>2</td>
<td>-</td>
<td>-</td>
</tr>
<tr>
<td>np</td>
<td>np</td>
<td>np</td>
<td>vp</td>
<td>-</td>
<td>-</td>
<td>0</td>
<td>3</td>
<td>2</td>
<td>1</td>
<td>-</td>
<td>-</td>
</tr>
<tr>
<td>np</td>
<td>np</td>
<td>vp</td>
<td>-</td>
<td>-</td>
<td>-</td>
<td>0</td>
<td>2</td>
<td>1</td>
<td>-</td>
<td>-</td>
<td>-</td>
</tr>
<tr>
<td>np</td>
<td>np</td>
<td>np</td>
<td>pp</td>
<td>vp</td>
<td>0</td>
<td>5</td>
<td>4</td>
<td>1</td>
<td>2</td>
<td>3</td>
<td>-</td>
</tr>
<tr>
<td>np</td>
<td>np</td>
<td>np</td>
<td>np</td>
<td>vp</td>
<td>-</td>
<td>0</td>
<td>4</td>
<td>1</td>
<td>2</td>
<td>3</td>
<td>-</td>
</tr>
<tr>
<td>np</td>
<td>np</td>
<td>vp</td>
<td>vp</td>
<td>-</td>
<td>-</td>
<td>0</td>
<td>3</td>
<td>1</td>
<td>2</td>
<td>-</td>
<td>-</td>
</tr>
<tr>
<td>np</td>
<td>pp</td>
<td>vp</td>
<td>vp</td>
<td>-</td>
<td>-</td>
<td>0</td>
<td>3</td>
<td>2</td>
<td>1</td>
<td>-</td>
<td>-</td>
</tr>
<tr>
<td>np</td>
<td>pp</td>
<td>np</td>
<td>vp</td>
<td>-</td>
<td>-</td>
<td>0</td>
<td>4</td>
<td>3</td>
<td>2</td>
<td>1</td>
<td>-</td>
</tr>
<tr>
<td>np</td>
<td>pp</td>
<td>vp</td>
<td>-</td>
<td>-</td>
<td>-</td>
<td>0</td>
<td>2</td>
<td>1</td>
<td>-</td>
<td>-</td>
<td>-</td>
</tr>
<tr>
<td>np</td>
<td>pp</td>
<td>np</td>
<td>vp</td>
<td>-</td>
<td>-</td>
<td>0</td>
<td>3</td>
<td>2</td>
<td>1</td>
<td>-</td>
<td>-</td>
</tr>
<tr>
<td>np</td>
<td>pp</td>
<td>pp</td>
<td>vp</td>
<td>-</td>
<td>-</td>
<td>0</td>
<td>3</td>
<td>2</td>
<td>1</td>
<td>-</td>
<td>-</td>
</tr>
<tr>
<td>np</td>
<td>vp</td>
<td>-</td>
<td>-</td>
<td>-</td>
<td>-</td>
<td>0</td>
<td>1</td>
<td>-</td>
<td>-</td>
<td>-</td>
<td>-</td>
</tr>
<tr>
<td>np</td>
<td>vp</td>
<td>np</td>
<td>vp</td>
<td>-</td>
<td>-</td>
<td>0</td>
<td>3</td>
<td>2</td>
<td>1</td>
<td>-</td>
<td>-</td>
</tr>
<tr>
<td>pp</td>
<td>np</td>
<td>vp</td>
<td>vp</td>
<td>-</td>
<td>-</td>
<td>1</td>
<td>3</td>
<td>2</td>
<td>0</td>
<td>-</td>
<td>-</td>
</tr>
<tr>
<td>pp</td>
<td>np</td>
<td>np</td>
<td>vp</td>
<td>-</td>
<td>-</td>
<td>1</td>
<td>3</td>
<td>0</td>
<td>2</td>
<td>-</td>
<td>-</td>
</tr>
<tr>
<td>pp</td>
<td>np</td>
<td>vp</td>
<td>-</td>
<td>-</td>
<td>-</td>
<td>1</td>
<td>2</td>
<td>0</td>
<td>-</td>
<td>-</td>
<td>-</td>
</tr>
<tr>
<td>pp</td>
<td>pp</td>
<td>np</td>
<td>vp</td>
<td>-</td>
<td>-</td>
<td>2</td>
<td>3</td>
<td>1</td>
<td>0</td>
<td>-</td>
<td>-</td>
</tr>
<tr>
<td>pp</td>
<td>vp</td>
<td>-</td>
<td>-</td>
<td>-</td>
<td>-</td>
<td>0</td>
<td>1</td>
<td>-</td>
<td>-</td>
<td>-</td>
<td>-</td>
</tr>
</tbody>
</table>
Here np refers to Noun Phrase, pp- Postpositional Phrase, vp for Verb Phrase.

The values in sen0, sen1, sen2 – etc. refer to the serial number of source language phrase. A total of 18 entries in the database, are derived, if the number of phrases is restricted to 6. The above entries signify the following rules for synthesis of phrases to form an English language sentence.

### 6.1.1 Description of Some Rules

<table>
<thead>
<tr>
<th>Punjabi</th>
<th>English</th>
</tr>
</thead>
<tbody>
<tr>
<td>NP1 + NP2 + PP + VP</td>
<td>NP1 + VP + NP2 + PP</td>
</tr>
<tr>
<td>NP + PP1 + PP2 + VP</td>
<td>NP + VP + PP2 + PP1</td>
</tr>
<tr>
<td>PP + NP1 + NP2 + VP</td>
<td>NP1 + VP + PP + NP2</td>
</tr>
<tr>
<td>NP + VP</td>
<td>NP + VP</td>
</tr>
<tr>
<td>PP + NP + VP</td>
<td>NP + VP + PP</td>
</tr>
<tr>
<td>PP1 + PP2 + NP + VP</td>
<td>NP + VP + PP1 + PP2</td>
</tr>
<tr>
<td>NP1 + NP2 + NP3 + VP</td>
<td>NP1 + VP + NP3 + NP2</td>
</tr>
<tr>
<td>NP1 + NP2 + VP</td>
<td>NP1 + VP + NP2</td>
</tr>
<tr>
<td>NP1 + NP2 + NP3 + NP4 + PP + VP</td>
<td>NP1 + VP + PP + NP2 + NP3 + NP4</td>
</tr>
<tr>
<td>NP1 + NP2 + NP3 + NP4 + VP</td>
<td>NP1 + VP + NP2 + NP3 + NP4</td>
</tr>
<tr>
<td>NP1 + NP2 + VP1 + VP2</td>
<td>NP1 + VP2 + NP2 + VP1</td>
</tr>
<tr>
<td>NP1 + NP2 + NP3 + VP</td>
<td>NP1 + VP + NP3 + NP2</td>
</tr>
</tbody>
</table>
6.1.2 Examples

P: उन्होंने पांच लाख रुपये मांगे 
T: uhnānē pañj lakh rupaē maṅgē san

E: They asked for five lakh rupees

P: उहदी भाड़ से सभी दे गई 
T: uhdī mātā vī zakhmī hō gaī

E: Her mother also got injured
6.2 Post Processing

Machine Translation (MT) keeps raising hopes and disillusionments. If readability and accuracy are to be guaranteed, it is necessary that the output of MT should be post edited. Post Processing is needed to increase the accuracy of the system. Since English and Punjabi languages have a great structural divergence, the output needs to be post edited to increase the accuracy of the system. There are a number of important grammatical divergences i.e. gender and number divergences which affect agreement. The grammar is incorrect or the relation of words in their reference to other words, or their dependence according to the sense is incorrect and needs to be adjusted.

This phase is the tail end of our Machine Translation System. It is a sentence level post-processing module that improves the translation quality by making corrections in the translation generated. In other words, it can be said that it is a system of correction for ill-formed sentences. The output generated by the
synthesis phase is the input for post-processing phase. This phase will correct the grammatical errors based on the rules implemented. In this section, we will discuss error categories introduced by the Machine Translation System, which leads to grammatically incorrect results, and thus needs to be corrected. It is not possible to fully remove all the grammatical errors, but accuracy can be improved to some extent.

6.2.1 Related Work

In 1988, Nishida and colleagues described a Post-Editing Correction information Feedback system (PECOF) in its early stages, which attempts to improve a Transfer-based MT system. Su et al. (1995) have explored the possibility of using feedback for a corpus-based MT system to adjust the system parameters so that the user style could be respected in the translation output. Menezes and Richardson and Imamura have proposed the use of reference translations to “clean” incorrect or redundant rules after automatic acquisition. [155]. Allen and Hogan explored a view to fix systematic errors committed by an MT system [43]. When these MT errors cannot be fixed with advanced User Dictionary coding techniques, they may be fixed using powerful global search and replace patterns. Roturier used regular expressions for post-processing module in Machine Translation System. [47] Kartunnens suggests applying finite automata and transducers that represent regular expressions, for natural language texts. [156]
Number of hybrid experiments have been conducted by combining rule-based MT (RBMT) systems with Statistical Post-Editing (SPE) systems.

6.2.2 Rules for Automatic Post Editing

Many papers on post editing state in various ways that post editors struggle constantly with the issue of the quantity of elements to change while also keeping the translated text at a certain level of quality. Many of the existing articles on the topic of post editing indicate that there is possibly a set of rules that can be followed for making the MT + PE process a cost-effective one, but how can these be grouped together and applied in an evaluation metric. Rather than devise still yet another set of rules based on our own PE experience, we have decided that it is possible, and more effective, to conduct corpus-driven tests in order to deduce sets of post editing rules that have been applied by MT post editors. The idea behind this development work is that if an MT system makes a particular error when translating a document, it is very likely that it will commit the same error each and every time the same set of conditions are presented.

1. Addition of Preposition ‘to’, ‘for’, ‘with’ with some words

The preposition ‘to’ is added in the translated sentences if:

There is no preposition after the word ‘came’ or ‘come’ followed by a constituent of a noun phrase.

P - ਕੁਝ ਲੋਕ ਸਾਡੇ ਘਰ ਆਏ

T - kujh lōk sāḍē ghar āē
Incorrect - Some people came our house

Correct - Some people came to our house

Similarly ‘to’ is added after the word ‘going’ which is followed by another verb.

Incorrect - He was going see marriage

Correct - He was going to see marriage

The word ‘to’ is also added with the word ‘brought’

For Example

Incorrect - He was brought hoispital afterwards

Correct - He was brought to hospital afterwards

Preposition ‘for’ is added after the word ‘asked’.

Incorrect - They asked five lakh rupees

Correct - They asked for five lakh rupees

Addition of ‘with’ with the word ‘collision’ or ‘collided’

If no preposition is present with the word ‘collided’, add the preposition ‘with’
T - uhnē mōtar savār nūṃ ṭakkar mār ditī

Incorrect- He collided motor rider

Correct - He collided with motor rider

Replacement of any preposition with ‘of’ if used with the word ‘informed’

P - मेंठु मटी छड़ा उग सातू बुवालिशा लाखे

T - mainūṃ sahī ḍhaṅg tōṃ jāṅū karāiā jāvē

Incorrect- I should be informed from correct method.

Correct - I should be informed of correct method.

2. Deletion of Preposition, if present with an adverb

If prepositions and adverbs together exist in the translated sentence, the preposition is deleted

For Example.

P - हैजर वापस हेंट जमागल लिङ्गवा लिङ्गा

T - uhnūṃ bāad vicc haspatāl liāndā giā

Incorrect - He was brought to hospital in afterwards

Correct - He was brought to hospital afterwards

The preposition ‘in’ is deleted as it exists with an adverb ‘afterwards’.

3. Replace ‘of’ with ‘to’ if it exists between two verbs.

For Example.

P - विमेंघत ते वेंट ली चौबिन्द्र रा जीजी

T - kisē maimbar nē rōkaṇ dī kōshish nā kīti
Incorrect – No member tried of stop

Correct – No member tried to stop

In both the sentences, ‘of’ between ‘threatened’ and ‘kill’ and between ‘tried’ and ‘stop’ is replaced with ‘to’.

4. **Combining ‘not’ and ‘any’ to ‘No’**

If in the sentence ‘no’ and ‘any’ both are present, ‘any’ is converted to ‘no’ and ‘not’ is removed.

Incorrect – Any member not tried to stop

Correct - No member tried to stop

Similarly ‘anyone’ and the negation is replaced with ‘no one’

Incorrect – Anyone was not injured.

Correct - No one was injured.

5. **Changing the position of the adverb ‘often’**

All the adverbs are placed after the verb in the target sentence, but if the adverb is ‘often’, its position is changed and should be placed before the verb.

Incorrect – He teases often his wife

Correct – He often teases his wife
6. **Replacement of article ‘a’ by ‘an’**

Article ‘a’ is replaced with the article ‘an’ if the following word starts with a vowel.

P - ਇਹ ਇਕ ਅਨਾਂ ਹੈ

T - ih vī ik aniāṁ hai

Incorrect - This is also a injustice

Correct - This is also an injustice

6.2.3 **Approach followed for Post Editing Module**

Post Editing module first needs to detect the errors and then correct those errors. The error typology has four classes into which users implicitly classify the errors.

**Missing Word:** A particular word may be missing in the sentence. For Example, In many sentences ‘to’, ‘for’ etc. are missing and it needs to be added after checking some conditions.

**Extra Word:** Some translated sentences have some extra words present in it, which should be deleted. For Example. In some sentences, prepositions are extra, those must be deleted.

**Wrong Word Order:** Most of the errors lie in this range. It can be corrected by interchanging words, or by changing the position of word in a phrase or by placing the word before or after any phrase.

**Word Replacement:** Some of the words are replaced with other words depending upon the contextual information.
These errors after detection are given as input to Automatic Rule Refiner in order to fix these according to the rules. Rules cannot be available for each and every error, but the translation can be improved to a certain level by fixing some of the errors.

The implementation of Automatic Post Editing tool needs to detect the errors by matching the words in the translated output with a given set of words. In Post Editing module, we also need to know about the tag of each word in the target language sentence. In implementation of this module, the POS tag of each word is also passed along with translated output. Pattern matching technique is used to search for a particular string in the translated output.

The analysis was done on 2000 sentences. It was found that 17.45% of the output text has been corrected grammatically using these rules. Table 6.2 shows various error categories in correcting the grammatical errors:

<table>
<thead>
<tr>
<th>S.No.</th>
<th>Grammatical Error Category</th>
<th>Percentage Distribution</th>
</tr>
</thead>
<tbody>
<tr>
<td>1.</td>
<td>Missing Word</td>
<td>54.23%</td>
</tr>
<tr>
<td>2.</td>
<td>Extra word</td>
<td>16.34%</td>
</tr>
<tr>
<td>3.</td>
<td>Word Replacement</td>
<td>20.44%</td>
</tr>
<tr>
<td>4.</td>
<td>Wrong Word Order</td>
<td>8.99%</td>
</tr>
</tbody>
</table>
6.3 Sample Translations

For illustration purpose, the Translation of a sentence in Punjabi is shown at different stages.

1. P: ਨੌਜਵਾਨ ਦੀ ਸੜਕ ਹਾਦਸੇ ਵਿਚ ਮੌਤ ਹੋ ਗਈ

   T: naujvān dī saṛak hādsē vic maut hō gaī

Pre Processing Phase:

- In the step of Text Normalization, the word ਵਿਚ (vic) is corrected to ਵੱਚ (vicc), if wrongly spelled.
- A collocation ਹੋ ਗਈ (hō gaī) is identified and converted to ਹੋਈ (hōī).
- No named Entity is found.

   After Pre processing, sentence is converted to

   P: ਨੌਜਵਾਨ ਦੀ ਸੜਕ ਹਾਦਸੇ ਵੱਚ ਮੌਤ ਹੋਈ

   T: naujvān dī saṛak hādsē vicc maut hōī

Tokenizer: It will generate the words ਨੌਜਵਾਨ (naujvān), ਦੀ (dī), ਸੜਕ (saṛak), ਹਾਦਸੇ (hādsē), ਵਿਚ (vicc), ਮੌਤ (maut), ਹੋਈ (hōī). These tokens are generated and passed to next phase of translation.

Morph Analyzer and Part of Speech Tagger:

- After collecting the morph information, each token is attached with tags

   ਨੌਜਵਾਨ (n-m- -s-d- - - - , n-m- -s-o- - - - , n-m- -p-d- - - - ) , ਦੀ (pos- - - - - - - - ) ,
   ਸੜਕ (n-f- -s-d- - - - , n-f- -s-o- - - - ) , ਹਾਦਸੇ (n-m- -p-d- - - - , n-m- -s-o- - - - ) ,
Ambiguity between tags is removed and each token is attached with one tag.

Phrase Chunker

It groups the tokens into phrases

NP ਨੌਜਵਾਨ ਦੀ (naujvān dī)

PP ਸੜਕ ਹਾਦਸੇ ਿਵੱਚ (saṛak hādsē vicc)

VP ਮੌਤ ਹੋਈ (maut hōī)

Translation Engine

The translation engine translates each word in a phrase and joins it according to rules of target language.

NP Youngman

PP in road accident

VP died
**Synthesizer**

It combines phrases of Punjabi using rules of English.

NP + PP + VP - > NP + VP + PP

The resultant English sentence is

E: Youngman died in road accident

**Post Processing**

No rules of post processing are applied on the result.

2.  
P: ਸੁਰੀੜਨਰ ਕੁਮਾਰ ਨੇ ਮੋਟਰ ਸਵਾਰ ਨੂੰ ਟਕਰ ਮਾਰ ਿਦਤੀ

T: surinder kumār nē mōṭar savār nūṁ ṭakar mār ditī

**Pre Processing Phase:**

- In the step of Text Normalization, the word ṭਕਰ (ṭakar) is corrected to ṭਕਰੰ (ṭakkar) and ਿਦਤੀ (ditī) is corrected to ਿਦੱਤੀ (dittī), as they wrongly spelled.

- A collocation ਮਾਰੀ (mārī) is identified and converted to ਮਾਰੀ (mārī).

- ਸੁਰੀੜਨਰ (surinder) is recognized as a Named Entity as it is followed by subcaste ਕੁਮਾਰ (kumār)

After Pre processing, sentence is converted to

P: ਸੁਰੀੜਨਰ ਕੁਮਾਰ ਨੇ ਮੋਟਰ ਸਵਾਰ ਨੂੰ ਟਕਰ ਮਾਰੀ

T: surinder kumār nē mōṭar savār nūṁ ṭakkar mārī
Tokenizer: It will generate the words ਸੁਰੀਂਦਰ (surinder), ਕੁਮਾਰ (kumār), ਨੇ (nē), ਮੋਟਰ (mōṭar) ਸਵਾਰ (savār), ਨੂੰ (nūṃ), ਟੱਕਰ (ṭakkar), ਮਾਰੀ (mārī) These tokens are generated and passed to the next phase of translation.

Morph Analyzer and Part of Speech Tagger:

• After collecting the morph information, each token is attached with tags
  ਸੁਰੀਂਦਰ (pnoun- - - - - - - ), ਕੁਮਾਰ (snoun- - - - - - - ), ਨੇ (ipo- - - - - - - ), ਮੋਟਰ (n-f - s-d- - - , n-f - s-o- - - ), ਸਵਾਰ (n-m - s-d- - - , n-m - s-o- - - , n-m - p-d- - - ), ਨੂੰ (ipo- - - - - - - ), ਟੱਕਰ (n-f - s-d- - - , n-f - s-o- - - , v-b-s-s- - f-x- ), ਮਾਰੀ (v-f-x-s- - x-p- - , cverb- - - - - - - )
  
• Ambiguity between tags is removed and each token is attached with one tag.
  ਸੁਰੀਂਦਰ (pnoun- - - - - - - ), ਕੁਮਾਰ (snoun- - - - - - - ), ਨੇ (ipo- - - - - - - ), ਮੋਟਰ (n-f - s-d- - - ), ਸਵਾਰ (n-m - s-o- - - ), ਨੂੰ (ipo- - - - - - - ), ਟੱਕਰ (n-f - s-d- - - ), ਮਾਰੀ (cverb- - - - - - - )

Phrase Chunker

It groups the tokens into phrases

NP - ਸੁਰੀਂਦਰ ਕੁਮਾਰ ਨੇ (surinder kumār nē)

NP - ਮੋਟਰ ਸਵਾਰ ਨੂੰ (mōṭar savār nūṃ)

VP - ਟੱਕਰ ਮਾਰੀ (ṭakkar mārī)
**Translation Engine**

The translation engine translates each word in a phrase and joins it according to rules of target language.

NP - Surinder Kumar

NP - motor rider

VP - collided

**Synthesizer**

It combines phrases of Punjabi using rules of English.

NP1 + NP2 + VP - > NP1 + VP + NP2

The resultant English sentence is

E: Surinder Kumar collided motor rider

**Post Processing**

The preposition ‘with’ is added with collided.

Incorrect - Surinder Kumar collided motor rider

Correct - Surinder Kumar collided with motor rider

6.4 Summary

This chapter discusses in detail the generation component. It describes the rules for synthesizing a sentence in target language and also the post editing applied to improve the output. The implementation of the complete system has been discussed along with an illustrative example. In the next chapters, we will discuss the evaluation and results of the Punjabi to English Machine Translation System.